# A SPECIFIC NETWORK LINK AND PATH LIKELIHOOD PREDICTION TOOL

THESIS

Gary K. Moy, Captain, USAF

AFIT/GCS/ENG/96D-21

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GCS/ENG/96D-21

# A SPECIFIC NETWORK LINK AND PATH LIKELIHOOD PREDICTION TOOL

THESIS

Gary K. Moy, Captain, USAF

AFIT/GCS/ENG/96D-21

The views expressed in this thesis are those of the author and so no reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GCS/ENG/96D-21

# A
# SPECIFIC NETWORK LINK
# AND
# PATH LIKELIHOOD PREDICTION TOOL

THESIS

Presented to the Faculty of the Graduate School of Engineering

Air Education and Training Command

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Computer Systems

Gary K. Moy, M.S

Captain, USAF

December 1996

Approved for public release; distribution unlimited

## Acknowledgments

I would like to foremost thank my fiancee for her unending patience, support and love. Without her tolerance and help, this work would not have been possible. I am privileged to fully assume my position as husband to her one month after AFIT graduation.

I am also indebted to my advisor, Dr. Richard A. Raines for his guidance, assistance and leadership. I would like to thank him and Dr. Yupo Chan for providing the thesis topic and the Generalized Network Analyzer (GNA) software package. I appreciation my committee for their time and effort in support of my work.

Next, I would like to thank Mr. Brad A. Smith of SOCHE. Had he not provided me additional programming and consultation of the HOOPS graphical system in the GNA software package, this work would have been much more "theoretical."

I also should not forget to thank Marshall Messamore for checking my draft while he was watching a sports game on the television. Finally, I would like to thank the rest my classmates. Without their consult, camaraderie, and encouragement, this work would have been much more difficult.

Gary K. Moy

## Table of Contents

# List of Figures

## List of Tables

AFIT/GCS/ENG/96D-21

## Abstract

Communications have always been a crucial part of any military operation. As the pace of warfare and the technological complexity of weaponry have increased, so has the need for rapid information to assess battlefield conditions. Message passing across a network of communication nodes allowed commanders to communicate with their forces. It is clear that an accurate prediction of communication usage through a network will provide commanders with useful intelligence of friendly and unfriendly activities. Providing a specific network link and path likelihood prediction tool gives strategic military commanders additional intelligence information and enables them to manage their limited resources more efficiently.

In this study, Dijkstra's algorithm has been modified to allow the Queueing Network Analyzer's (QNA) analysis output to act as a node's goodness metric. QNA's calculation of the expected Total Sojourn Time for the completion of queueing and service in a node provides accurate measurement of expected congestion. The modified Dijkstra's algorithm in the Generalized Network Analyzer (GNA) is verified and empirically validated to properly deliver traffic. It appropriately generates the fastest traffic path from a start node to a destination node. This implementation includes notification if input parameters exceed the network's processing capability. GNA's Congestion Control displays notification and informs the user certain network input parameters must be lowered (PTR or BSTR) or where certain nodes must be improved to maintain node stability. With this unstable node identification, users can determine which node needs attention and improvements. Once this instability is removed, a good QoS is achieved and analysis proceeds. Upon successful completion of the analysis, GNA appends the generated route and expected sojourn time to a text file for later analysis. Use of two analysis techniques show the percentage of link usage within the 25 node test network. Three analytical techniques are provided to estimate the probable bounds of the input parameters and sojourn times. Using these techniques, a bound of the Total Sojourn Times is provided for a 16 node test network.

Given few input parameters, networks analyzed can provide a specific link usage probability and path likelihood. Since QNA requires few calculations and GNA's Congestion Control provides unstable node identification, using the given bound estimation techniques, designers and engineers can evaluate network topologies much more easily.

# A

# SPECIFIC NETWORK LINK

# AND

# PATH LIKELIHOOD PREDICTION TOOL

## *I. Introduction*

In the past several decades, the military has relied on a combination of land line phone systems and line-of-sight communications. Communications have always been a crucial part of any military operation. As the pace of warfare and the technological complexity of weaponry have increased, so has the need for rapid information to assess battlefield conditions.

The above situation was epitomized by the Gulf War, where the pace of the ground war moved quickly. Once land based communication nodes were established, satellites were able to assist in linking the command structure to the forces deployed. At the peak of operations, 700,000 phone calls and 152,000 traffic messages were processed per day, 5% over NATO assets, and 20% over commercial satellites [TuA93]. Message passing across a network of communication nodes allowed commanders to communicate with their forces.

## *1.1 Significance*

As mentioned, the fast pace of warfare dictates the need for rapid information to assess battlefield conditions. Additional intelligence information can always aid a leader in making decisions. Providing a specific network link and path likelihood prediction tool gives strategic military commanders additional intelligence information. This information is a prediction of friendly and unfriendly communications usage. Such information can provide decision makers, at various levels of command, insight into a variety of communication concerns. For example, two exposed, above ground sites within the network may have a high likelihood of passing information between the command staff and their units. These two sites and the links used need to be protected from unfriendly disruptions. Knowledge of which specific network link is in use enables the decision makers to manage their limited resources more efficiently.

## *1.2 Background*

***1.2.1 Network Virtual Path Selection Strategies.*** Virtual path routing is a method of packet switching, which is one of the two major forms of message passing across a network. All virtual path routing methods used for packet switching follow a virtual path selection algorithm. This algorithm selects the path from the source node to the sink node for use in passing all the messages across the network. Most path selection algorithms attempt to find the most efficient path from source to sink. An efficient path can be defined in a number of ways, depending on the network in question.

***1.2.2 Generalized Network Analyzer.*** The Generalized Network Analyzer (GNA) is a network prediction tool comprised of several general and detailed network models. Taking advantage of general and detailed models, GNA can be used at different levels of decision making. Currently, GNA algorithms are based on circuit switching, one of the two popular routing methods. The other popular algorithm GNA does not support is packet switching. With an additional routing algorithm integrated, GNA can become a useful tool in determining specific link and path likelihood prediction.

***1.2.3 Queuing Network Analyzer.*** The Queuing Network Analyzer (QNA) is a model within the GNA. QNA analyzes open networks of queues. The general approach in QNA is to characterize the arrival process by two to three parameters. QNA will then analyze each of the involved nodes separately. The packet switching subsystem can use the QNA model.

## *1.3 Research Problem*

Any information that clarifies a battlefield situation is useful. This research seeks to integrate an efficient packet switching algorithm model into GNA and allow it to be used as a specific network link and path likelihood prediction tool.

***1.3.1 Constraints.*** This research is constrained by two factors. The first factor is the routing algorithm chosen for formulation. These algorithms are limited to static and adaptive techniques. Secondly, node and link reliability is not considered to be a factor. The scope of this research takes into account a wide area network (WAN) topology that remains constant once it is generated.

***1.3.2 Sub-objectives.*** This research concentrates on determining the feasibility of integrating packet switching formulations into the GNA for use as specific network link and path likelihood prediction tool. The following are sub-objectives supporting this thesis:

1. Identify resource constraints

2. Identify the candidate packet switching algorithms for use in this research

3. Identify the constraints and limitations of the current GNA software

4. Identify the elements needed for implementing a packet switching subsystem within QNA

5. Integrate the packet switching subsystem into GNA

6. Examine the robustness of the packet switching subsystem within GNA.

### *1.4 Methodology*

The first sub-objective is realized by investigating the current maintainer's development plans for the GNA package. Additional resources can aid the development of the prediction tool. In order to provide the user with a complete, self-contained package, the approval of additional network tools must be given.

The second sub-objective is met by continued literature review. The goal focuses on determining the best routing algorithm to integrate as a packet switching subsystem into GNA. This sub-objective is subject to scope and assumptions of the research. The third sub-objective is accomplished by detailed examination of the models present within GNA. The first section, the GNA software, is examined. The second section, FORMULA and Queuing Network Analyzer (QNA), is also examined. The work done in sub-objectives two and three are documented in Chapter Two of this thesis.

The fourth sub-objective is met by identifying the packet switching parameters parallel to the circuit switching parameters within QNA. The fifth sub-objective is completed after the data representation of the algorithm is transformed accurately into the data representation of the GNA models.

For the sixth sub-objective, experiments are conducted and data will be collected. A series of experiments are constructed to verify and validate the integrated packet switching algorithm within GNA. The new packet switching subsystem is tested on sample networks. Each test experiment has results that are compared to standard theoretical results. Work progresses as follows:

Conduct Experiment

a. Build the necessary simulations to test the specific experiment

b. Run the test cases defined in the previous section

c. Collect relevant data

d. Validate simulation results for correctness

Analyze Results: Analyze data from the experiment to evaluate the tool

a. Plot data from experiments, using the metrics developed

b. Compare actual results versus theoretical results

c. Analyze data and summarize results

The experiments are conducted and results analyzed as described above. The work done for sub-objectives four, five, and six are documented in Chapter Three and Four of this thesis.

# *II. Literature Review*

## *2.1 Introduction*

The fast pace of warfare dictates the need for rapid information to assess battlefield conditions. Any information that clarifies a battlefield situation is useful. Additional intelligence information can always aid a leader in making decisions. An accurate prediction of communication usage through a network provides commanders with useful intelligence of friendly and unfriendly activities. Providing a specific link and path likelihood prediction tool gives strategic military commanders additional intelligence information. This information is a prediction of friendly and unfriendly communications usage. Generalized Network Analyzer (GNA) can provide decision makers, at various levels of command, insight into a variety of communication concerns [RoP91]. Integrating an efficient packet switching algorithm model into GNA allows it to be used as a specific network link and path likelihood prediction tool.

This chapter presents a discussion of the background of GNA, a summary review of possible routing techniques and algorithms to be implemented in GNA. Section 2.2 describes the framework of the GNA along with a summary of the two models within it. A discussion of how each model can represent a network comprises Section 2.3. A review of possible packet switching routing algorithms as they are applied today are introduced in Section 2.4. In closing, section 2.5 summarizes the information covered in this chapter.

## *2.2 Generalized Network Analyzer (GNA)*

GNA is comprised of several analytical network models and conversions between them [DaN94]. There are commonalties between the data representations needed for different models. The GNA algorithms exploit these commonalties. For example, GNA allows users to concentrate on understanding the detailed workings of a queueing system, on improving it, or a combination of both. Users can analyze a range of network systems instead of being restricted to a limited subset. Transformation between models permit verification of the same problem using several models and can provide insight into problems and models in general [RoP91]. Taking advantage of general and detailed models, one can appeal to different levels of decision making [RoP91]. For example, managers may be mostly interested in the cost to optimize a network while network support engineers may be mainly interested in the details of network performance, such as queue lengths.

GNA greatly simplifies experiments by automating the tasks of data input, computational analysis, conversion of data between models, and the display of model output. Output has only to be analyzed and compared visually by inspecting graphical output on-screen and from print-screens. Symbolic representations of network components are used so that decision makers will readily understand the results of the analysis. Suggested improvements to network components can be discarded or made part of the current network's topology. After the network is modified, it can be reanalyzed. This process can continue until a predetermined goal is achieved, or until all alternatives have been exhausted. This cyclic approach can be used for network analysis [Mel86].

*2.2.1 GNA Notation.* The following notation is used which allows the two models to share common data representation:

$c_i$ = capacity of node *i*

$p_i$ = reliability of node *i*

$f_j$ = flow through path *j*

$R_j$ = reliability of path *j*

$p'_k$ = reliability of arc *k* (substitute *k* = (i,j) to make obvious the origin and terminal nodes of the arc)

$c'_k$ = capacity of arc *k* (substitute *k* = (i,j) to make obvious the origin and terminal nodes of the arc)

$[a_{ij}]$= adjacency matrix where $a_{ij} = 1$ if node *i* connects to node *j*, otherwise $a_{ij} = 0$

$\tau_i$ = mean service time at queueing node *i*

$\mu_i$ = mean service time at queueing node *i*

$c^2_{ski}$ = variability parameter of the service-time distribution of customer class (route) *k* at the *i* th node of its route.

$c^2_{si}$ = variability parameter of the service-time distribution of all customer classes at the *i* th node of its route.

$\lambda_{0i}$ = external arrival rate to node *i*

$c^2_{0i}$ = variability parameter of the external arrival process to node *i*

$\hat{\lambda}_k$ = $\hat{\lambda}$ = external arrival rate of the system

$c^2_k$ = $c^2$ = variability parameter of the external arrival process of the system

$\lambda_j$ = total arrival rate at node *j*

$\gamma_j$ = multiplicative factor of customer creation for node *j*

$q_{ij}$ = probability that a customer will go from node *i* to *j*, the routing matrix

$\rho_i$ = traffic intensity at node *i*

$d_i$ = departure rate from node *i*

***2.2.2 GNA Operational Notes.*** There are some items of note within the GNA package. GNA's icon display is internally preset to a fixed size and location. Developing a network with a large number of nodes, as in this study, will cause it to inherently appear crowded. An option to resize the network is available. All nodes of the network can appear smaller or larger as the user desires. When the network size is reduced, any additional creation of nodes will appear in the original size. Therefore, some nodes will appear smaller and some nodes larger. Furthermore, modification to existing nodes will often occur as the user makes changes to obtain a good Quality of Service (QoS) for the network. This entails improving the process capability of each server and/or increasing the capacity of the node. Once the details are changed, a label is placed to identify the node and the new information. If resizing has occurred, the label may not appear in the intended location. It can appear away from the original node, over the node itself, or even over another node.

GNA maintains a data structure for use by the different models. Any modifications to the nodes of the network must be preceded by a clean load of the network file. The clean loading ensures the data structure values do not contain unintended changes before network modification begins. Once modifications are completed, saving the network to a new file is highly recommended. Performing a clean load of the new file before running this study's model will ensure an accurate input of network node and route data.

*2.2.3 **The Prescriptive Model.*** FORMULA, the more general of the two models, was originally designed for investigating the behavior of stochastic communication networks. It is used to determine expected network performance (throughput and reliability), congestion points, and how throughput and network reliability can be improved by increasing the capacity and/or reliability of network components, based on a budget available for improvement. FORMULA is a model taking into account both network analysis and synthesis. Since large stochastic networks are characterized by a huge number of failure states, calculating the expected maximum flow [MoE78] is computationally infeasible for even moderately sized networks [Yim88], as is the calculation of reliability measures in stochastic networks [Bal80]. FORMULA considers the lower and upper bounds of expected maximum flow to make approximation of network performance feasible [Yim88]. For example, the calculation of exact expected maximum flow requires the enumeration of all possible network states/topologies while the lower bound doesn't -- the lower bound formulation assumes no rerouting while the expected maximum flow formulation does. Using bounds as approximation, large networks can be analyzed and improved easily because the program creates network formulations and solutions very quickly. Detailed simulation results validated this approach [Yim88].

FORMULA represents a network as a group of arcs and nodes through which some commodity flows from a single source (the start of flow) to a single sink (the end of flow) along a finite number of paths. Arcs and nodes are treated the same in FORMULA, so it refers to both as "components." Components are assumed to operate independently of each other and can exist in either of two states, operating or not operating. FORMULA could be used to model the flow of electronic packets of information (commodity) through cables (arcs) and computers (nodes), where nodes and arcs are subject to both failure and capacity constraints. FORMULA describes each network arc or node $i$ with two values: capacity ($c_i$) and reliability ($p_i$). Reliability is the probability that a component will be operating at a given point in time, FORMULA finds all the paths through the network and determines the probability that each is operating, and the flow through each. A path $A_i$ is described with two values: flow ($f_i$) and reliability ($R_i$). Since FORMULA assumes that any arc, node, or path exists in either of two states, working or not working, each network arc, node, or path has two data points that describe it. There is a state where the component is working ($p_i$, or $p_k'$) with a known capacity ($c_i > 0$ or $c_k' > 0$) or flow ($f_i$) and a state where the component fails ($1-p_i$ or $1-p_i'$) or path fails ($1-R_i$) in which case the capacity or flow is zero.

Once FORMULA finds all paths and their reliabilities, it then formulates an objective function, to be minimized or maximized, with constraints. Several linear models and one non-linear model of the network can be generated. Note that in the equations below, $p_i$ and $c_i$ apply to both nodes and arcs (even though users sometimes distinguish between arcs and nodes.) The lower bound formulation [AnN80, Yim88], a linear formulation for approximating the expected maximum flow throughput is shown below.

The reliability of path $A_i$ is

$$R_j = \prod_{i \in A_j} p_i \tag{2.1}$$

The sum of expected flows on all paths $q$ from source $s$ to sink $t$ is given by

$$\sum_{j=1}^{q} R_j f_j \tag{2.2}$$

With no rerouting, the lower bound throughput can be formulated as

$$Max \sum_{j=1}^{q} R_j f_j \tag{2.3}$$

such that

$$\sum_{j=1}^{q} a_{ij} f_j \leq c_i \tag{2.4}$$

for $i = 1, .., n$ where $n$ is the total number of arcs and nodes, $q$ the total number of paths, and $f_j \geq 0$. The formulation for the upper bound of the expected maximum flow [Yim88] is

$$Max \sum_{j=1}^{q} f_j \tag{2.5}$$

such that

$$\sum_{j=1}^{q} a_{ij} f_j \leq E[c_i] \tag{2.6}$$

for $i = 1, ..., n$ where $n$ is the total number of arcs and nodes, $q$ the total number of paths, and $f_j \geq 0$ and E $[c_i]$ is the expected capacity of component $i$.

***2.2.4 The Descriptive Model.*** Queueing Network Analyzer (QNA), the more detailed model, was designed for the performance analysis of queueing networks. It can analyze open networks of queues with the first-come, first-serve discipline and no constraints on capacity. Each queueing node can have multiple servers. QNA calculates approximate measures of network congestion like expected queue lengths, throughput, traffic intensities, and expected sojourn time. In QNA, service-time distributions do not have to be exponential and external arrival processes do not have to be Poisson. QNA can operate on non-Markovian networks [Whi83a]. The model has been used to study communication networks and computer systems. It has produced good estimations compared to simulations and other approximations (M/M/1, M/G/1, and GI/G/1) of queueing network performance [Whi83a].

QNA can describe a network in terms of queueing nodes, customers, and fixed routes. The customers represent the commodity flowing through the network. Each customer class follows a fixed route (a set of nodes sequences), or a routing matrix representing random routing to flow through the network. Customers receive some kind of service at each node and nodes can have any number of servers. Customers enter the system from an external node onto directed arcs, move from node to node along the internal arcs, and ultimately leave the network on one of those arcs from an internal node to the external node. Customer traffic on the arcs are assumed to be random, and so are represented as stochastic processes. In this study, QNA can also be used to model the flow of electronic packets of information (commodity) through cables (routes) and computers (nodes), as in FORMULA. QNA cannot address reliability except through the use of the mean service time, but it can provide more specific information about network performance like the average number of packets of information waiting in a computer's buffer.

Each queueing node in QNA is described by its mean service time ($\tau_i$), a variability parameter ($c^2_{ski}$) describing the variance of the service time, its external arrival rate $\lambda_{0i}$ , and a variability parameter ($c^2_{0i}$) describing the variability of the arrival process. Since service time variability data was input for each node during the study, all customer classes $k$ have the same distribution for service time at each visit to node $i$ [Whi83b], the notation ($c_{si}^2$) can be used, leaving out the $k$. Each route (customer class) through the network is described by an external arrival rate ($\hat{\lambda}_k$), and a variability parameter ($c_k^2$) describing the variance of the arrival process. The notation $\hat{\lambda}$, and $c^2$ are used for clarity.

*2.2.5 **Queuing Network Analyzer Algorithm.*** Since this study concentrates on QNA, a summary look at QNA's analysis is needed. A detailed examination of the analysis is presented in Chapter Three of this study. There are three main steps in the QNA analysis: first, systems of linear equations are solved to find the internal arrival processes' parameters; second, each queue is analyzed as a standard G/G/m model as a renewal arrival process where each queue is characterized partially by the first two moments of the interarrival-time and service-time distributions; and finally, measures of congestion for the whole network are obtained by assuming, as an approximation, that the separate queues are stochastically independent given the flow parameters [Whi83b]. Large queueing networks can be analyzed quickly because of the minimal calculations necessary [Whi83b].

## *2.3 GNA Formulation*

Both models can represent networks in different ways. Using the prescriptive model, one can model a network as stochastic, where network components are susceptible to failure, or as deterministic, where all components are one-hundred-percent reliable. The descriptive model can be used to represent a network in which routing is random or in which is deterministic. The flexibility provided by the two models and the flexibility inherent in GNA's architecture allows great versatility in analysis of the same networks using several representations. Both models have analogous parameters for describing flow, but treat flow at different levels of detail. In the prescriptive model, commodity flow at components is limited by capacity. This defines the maximum numbers of units of commodity that can pass through the component per unit time. In the descriptive model, commodity flow at queueing nodes is quantified by mean service rates, the average number of units of commodity that can pass through the node per unit of time. Mean flow, aggregated from capacity and reliability data in the prescriptive model, can be directly analogous to mean service rate.

Another analogy is between mean flow on paths in the prescriptive model and arrival rates to routes in the descriptive model. Mean flow along a path in the prescriptive model, aggregated from flow and reliability data, describes the average number of units of commodity that traverse that path per unit time. The law of conservation of flow [JeB80] applies to all arcs/nodes in the prescriptive model. That is, the amount of flow entering a component must equal that leaving it. Since all flow begins at a single source and terminate at a single sink and there is conservation of flow, then flow at the source equals the flow at the sink. Stated another way, the amount of commodity entering the network equals that leaving it. At least the mean flow at the source and sink are known. The arrival rate to a route in the descriptive model describes the average number of units of commodity that arrive to the first node on the route from the outside world; this route has a deterministic node sequence along which commodity flows through the network. In general, a multi-commodity flow formulation for the prescriptive model will have an arrival rate for each source-sink pair. There can be several source-sink pairs. In an equivalent way, the descriptive model supports any number of customer classes. Another analogy is that between arcs and routing matrix entries, $q_{ij}$. Just as an arc (*i,j*) links two nodes in the prescriptive model, $q_{ij} \neq 0.0$ specifies a path for flow between queues *i* and *j* in the descriptive model.

Several network performance measures are used when comparing the two models. In the descriptive model, traffic intensity is a measure of resource utilization [Whi83b] or, "the ratio of the rate at which 'work' enters the system to the maximum rate (capacity) at which the system can perform this work" [Kle75]. This parameter can take on a value from zero to one -- zero being the lowest utilization, one the highest. So a queue with high traffic-intensity is identified with congestion, just as is a bottleneck in the prescriptive model. Both models calculate total throughput through a network, this value is used as another parameter for comparison.

## *2.4 Routing Algorithms*

Virtual path routing is a method of packet switching, which is one of the two major forms of message passing across a network. All virtual path routing methods used for packet switching follow a virtual path selection algorithm. This algorithm selects the path from the source node to the sink node for use in passing all the messages across the network. Most path selection algorithms attempt to find the most efficient path from source to sink. An efficient path can be defined in a number of ways, depending on the network in question.

Rather than exploring in depth all the different algorithms used in all types of networks, a following summary discussion of a number of routing algorithms with various applications are examined.

***2.4.1 Improved Routing Tree.*** The analysis of Elmore delay in distributed Resistance-Capacitor (RC) tree structures show the influence of both tree cost and tree radius on signal delay in VLSI interconnects. It gives a new and efficient interconnection tree constructions. This new construction smoothly combines the minimum cost and the minimum radius objective. This is accomplished by combining respectively optimal algorithms due to Prim and Dijkstra. Previous "shallow-light" techniques [AwB90, CoK92, KhR93] are both less direct and less effective. In practice, these methods achieve uniformly superior cost-radius tradeoffs. Timing simulations for a range of Integrated Circuitry (IC) and Multi-Chip Module (MCM) interconnect technologies show that the wire length savings yield reduced signal delay when compared to shallow-light or standard minimum spanning tree and Steiner tree routing [AlH95].

***2.4.2 Spanning Tree.*** Given an analysis of the performance of local area networks (LAN) interconnected according to the spanning tree and source routing techniques proposed in the framework of the IEEE 802 committee, two LAN interconnection schemes are presented in simulation models. Indications on how the performances of the interconnected systems are dependent on the placement and configuration of the interconnection devices (bridges). The two LAN interconnection techniques are analyzed and compared by means of curve plotting performance figures. An example is the frame delay or the network throughput, as a function of the total traffic offered to the interconnection system. Some approximated analytical results are also presented.

The authors, Munafo et al., showed that the spanning tree technique has frame delays that are dependent upon the location of the bridge. While the maximum throughput became higher because of the lower congestion of the LAN's proximity to the root of the spanning tree. The source routing technique showed increased performance for increased connectivity levels, but the gains was not as large as expected [MuN94].

***2.4.3 Optimal Minimax Routing.*** In Open Shortest Path First (OSPF) routing, the network is modeled as a graph and each link is associated with a non-negative arc weight (or link set metrics). A shortest path spanning tree is calculated for each origin to carry both the individually addressed and the group addressed (multicast) traffic. The OSPF routing protocol is adopted as a major part of the default Inter-Switching System Interface (ISSI) routing algorithm for Switched Multi-megabit Data Service (SMDS) networks where arc weights are inversely proportional to the aggregate link set capacities. The technique involves choosing a set of link set metrics so that the maximum link utilization factor is minimized in an SMDS network.

The situation is a difficult problem to solve optimally. The authors, Lin and Wang, approach this problem by proposing two algorithms on eight test networks. Their first approach uses the Lagrangean relaxation technique. This allows the solution to be decomposed into three smaller sub-problems. Their second approach is compromised of a seven step algorithm with two main advantages. First, their algorithm is simple and therefore may allow a larger number of iterations or shorter computational requirements. Second, individually addressed traffic and multicast traffic are considered in a uniform way. This reduces the number of iterations or computations required. The end solutions allowed improvements from 7% to 133% of maximum link utilization over the network's default routing method [LiW93a].

***2.4.4 Single-Path Minimax Routing.*** Consider the routing problem in networks with single-path routing and multicast services. In such networks, all the single-destination traffic is transmitted over exactly one path between the origin and the destination and the multiple destination traffic is transmitted over exactly one tree rooted at the origin. Examples of such networks are the conventional circuit-switched telephone networks, virtual-circuit based packet networks such as the X.25 networks, networks supporting SMDS, networks supporting Frame Relay Service (FRS), and the Broadband Integrated Services Digital Networks (B-ISDNs) based on Asynchronous Transfer Mode (ATM). Consider choosing a route/tree between every origin and its single/multiple destinations(s) in a network so as to minimize the maximum link utilization.

Lin and Wang approached the algorithm by using the Lagrangean relaxation. Their algorithm decomposed the problem into two smaller, independent, more easily solvable sub-problems. Since their algorithm is not require to be performed in real time, the computational complexity is not a major concern. The computational results show their solution to be near optimal (within 2%) in networks with up to 61 nodes. The range of improvements is from 16.67% to 75.00% over the standard minimum hop routing scheme [LiW93b].

***2.4.5 Optimal Minimax Routing in ATM Networks.*** Asynchronous Transfer Mode (ATM) has been adopted by the CCITT as the transport mode in which Broadband ISDN will be based. The flow model presented routed cells in an ATM network measuring network performance such as end-to-end delay and throughput. The objective is to minimize the largest cell loss probability among all links. The constraints correspond to a multi-commodity network flow problem with gains. The minimax routing algorithm determines a global optimal flow assignment.

Yee and Lee's algorithm was implemented and tested on several networks. Computational experiments showed that after repeated iteration, the algorithm reduces geometrically the interval containing the largest cell lose probability. Their algorithm converges to an optimum solution [LeY94].

***2.4.6 Optimal Hierarchical Routing Strategy.*** Saha and Mukherjee showed the computational analysis of an optimal hierarchical routing to choose the best communication path in a large network. Their scheme can select the optimal route in a hierarchical computer communication network. The solution of optimal routing in a hierarchical network is formulated as a non-linear combinatorial problem solved using the Lagrangean relaxation and subgradient optimization techniques. Saha and Mukherjee presented a simulation study on the application of the technique to a number of networks. Although their results provided validation of their technique, it also showed the higher the computational cost parameter for each hierarchical level, the higher the total design cost for the route computation [SaM95].

*2.4.7 Hierarchical Adaptive Routing.* Adaptive routing can improve network performance and fault-tolerance by providing multiple routing paths. However, the implementation complexity of adaptive routing can be significant, discouraging its use in commercial, massively parallel systems [Chi93]. Hierarchical Adaptive Routing (HAR) introduces a new adaptive routing framework, which provides a unified framework for simple and high performance, fully-adaptive deadlock-free wormhole routing. HAR divides the physical network into several levels of virtual network. There is one connection channel between two adjacent virtual networks that allows blocked packets in the higher level to move to the lower level. Different routing algorithms can be used in each virtual network, and the overall network is deadlock-free provided the routing algorithm in the lowest level virtual network is deadlock-free.

However, the routing algorithm in any other virtual network can be fully adaptive, even non-minimal to increase performance. HAR has three advantages: fully-adaptive deadlock-free routing in any non-wrapped and wrapped k-ary n-cube network with two or three virtual channels respectively, relatively small crossbars, and applicability to a wide variety of network topologies. Liu and Chien, the authors, provided detailed implementation and simulation studies of a HAR for 2D mesh networks [LiC94].

*2.4.8 Virtual Path and Virtual Circuit Routing.* The use of virtual paths in ATM networks reduces the call set-up delays, simplifies the hardware in the transit nodes and provides simple virtual circuit admission control. However, it also reduces the degree of capacity sharing and, therefore increases the call blocking rate. Given a network topology, link capacity of each physical link and traffic requirement of each origin-destination pair, the following four variables can minimize the expected call blocking rate subject to call set-up time constraints:

1. The node pairs that should have virtual paths.
2. The route for each virtual path.
3. The bandwidth assigned to each virtual path.
4. The routing assignment for each virtual circuit.

The solution is found as a nonlinear, nondifferentiable combinatorial optimization problem. The solution is an efficient two-phase procedure. The capacity of each virtual path is fixed and the routing assignments are adjusted to reduce the overall call blocking rate. In the second phase, the virtual path capacity assignments are adjusted to reduce the overall call blocking rate, using the fixed virtual circuit routing assignment from the first phase. The macro logic link concept is defined to simplify the assignments. The two phases are iterated until no improvements to the overall call blocking rate can be made [ChL95].

## *2.5 Summary*

This chapter discussed the framework of the General Network Analyzer, then presented key aspects of the prescriptive model, FORMULA, and the descriptive model Queueing Network Analyzer (QNA). This discussion was aimed at providing a background of the models used. In Section 2.3, an overview of the formulations of GNA was given to show how the two models can be used to describe a network. A review of several possible routing algorithms for use was presented in Section 2.4. This review examined representative samples of routing algorithms used for network communication. The background information on GNA and in particular QNA, and the selected routing algorithm for inclusion into QNA will be explored more thoroughly in Chapter Three.

THIS PAGE INTENTIONALLY LEFT BLANK.

# *III. Methodology*

## *3.1 Introduction*

This chapter discusses the methodology used to produce the GNA Specific Network Link and Path Likelihood Prediction Tool. After the introduction, this chapter is divided into seven sections to discuss the procedures applied by the GNA tool. The first section is intended to give the reader an elementary overview on the usage procedures of the GNA tool. System model operating assumptions are included in this discussion. The next four sections explain each procedure and discuss the principles behind their use. The Packet Switching Network section examines the development of a network within GNA. The Route Determination section explains the operational reasons for the chosen routing procedure. The Generalized Network Analyzer Routing Algorithm section verifies and empirically validates the routing algorithm within GNA. The Queueing Network Analyzer section explores in detail the queueing method found in GNA as developed by Whitt. The Path Likelihood section introduces the statistical analysis that will be used to determine which links within the network are most likely to be in use. The seventh section summarizes the important aspects of this chapter.

### *3.2 Elementary Overview*

Initially the user graphically creates, through the GNA interface, a network for use in the packet switching environment. GNA requests a few basic parameters to describe the network servers, traffic and input/output characteristics. GNA then initiates QNA to estimate the expected traffic within the network. Based on this traffic estimation, a routing algorithm generates a path within the network from source to destination. This path will be the minimum traffic path through the network. GNA provides QNA with this path to estimate network performance. The user repeats these steps with varying conditions to produce a statistical analysis of the most likely set of links used.

This study operates above the physical and data link layers of the standard Open System Interconnect (OSI) model. Therefore, it is assumed that the mechanical, electrical, functional, and procedural functions required to establish, maintain, and release the physical connections between the data terminal equipment and data circuit-terminating equipment of the nodes and the attached hosts are operating. In addition, the network created has the fundamental structure of the data link control protocols. Aspects such as bit-oriented frame control, error control, initialization control, link management capabilities, transparency, flow control, and abnormal recovery control are inherently working within the GNA model created by the user. Given these assumptions, the terms route and path are used interchangeably. A route or path refers to a link or a set of links between nodes, where each link connects only two nodes. References to frame, packet, message, traffic, or commodity all refer to the same thing.

This study concentrates on a single form of traffic, even though the study differentiates the primary customer's traffic from the remaining traffic in the network. Therefore, there exists a basic rate to characterize the amount of traffic each node expects. Additionally, each node's processing abilities and characteristics are assumed to be exponentially distributed regardless of the packet it receives. This decision is based on the modeling of the network above the data link control layer. Once a server at a node receives a packet, the server cannot differentiate between the primary customer and a secondary customer until the server is processing the packet. Therefore, any packets arriving by an incoming link to a particular node requires the same mean amount of processing time. This is not to say that all nodes are identical. The user should model the network to reflect the higher capabilities of certain nodes which are deemed important.

### *3.3 Packet Switching Network*

***3.3.1 Network Topology.*** The user initially creates a network topology within GNA to include all the participating nodes and the existing links between the nodes. The user includes, for each participating node, the service time required to process any single incoming packet to determine its destination, and the number of servers in the node. GNA assumes the attached hosts or systems to a node are always present. GNA does not require the host's graphical icon in order to eliminate additional clutter. Once all the nodes are defined, a link is placed between every pair of nodes where a connection exists. It should be noted that GNA accepts one pair of nodes per attempt to place a link. For example, if links exist from node one to node two and from node two to node three, this would constitute one attempt with nodes one and two and a separate attempt with nodes two and three.

The designed test network topology is based upon a military tactical setup of 25 sites. The network topology is based upon a military Lines of Communication (LOC) deployment [Dep84]. A campaign or major operation should never depend on a single LOC. LOCs are the routes that connect an operating force with a base of operations and along which supplies and military forces move. Relying on one LOC results in competition by various land, naval, and air forces. The competition for a single LOC causes bottlenecks lowering effectiveness and efficiency for distribution of logistics resources [Dep93]. Moreover, where austere logistics resources limit multiple LOCs, security of air and ground LOCs are particularly important. Protecting LOCs at a minimum cost to committed combat units includes using geographic features, friendly civil security forces, and uncommitted combat units. It may be necessary to conduct a major operation to secure the LOCs required to support later phases of a campaign [Dep93]. Shown in Figure 3.1 is a typical command post composition in a control measure commonly employed in the defense [Dep84, Dep85]. The network topology used in this study, as shown in Figure 3.2a and Figure 3.2b, is based on the Figure 3.1. Figure 3.2a shows the links existing between a node and a communications relay node. To avoid clutter, Figure 3.2b shows the remaining existing links not shown in Figure 3.2a.

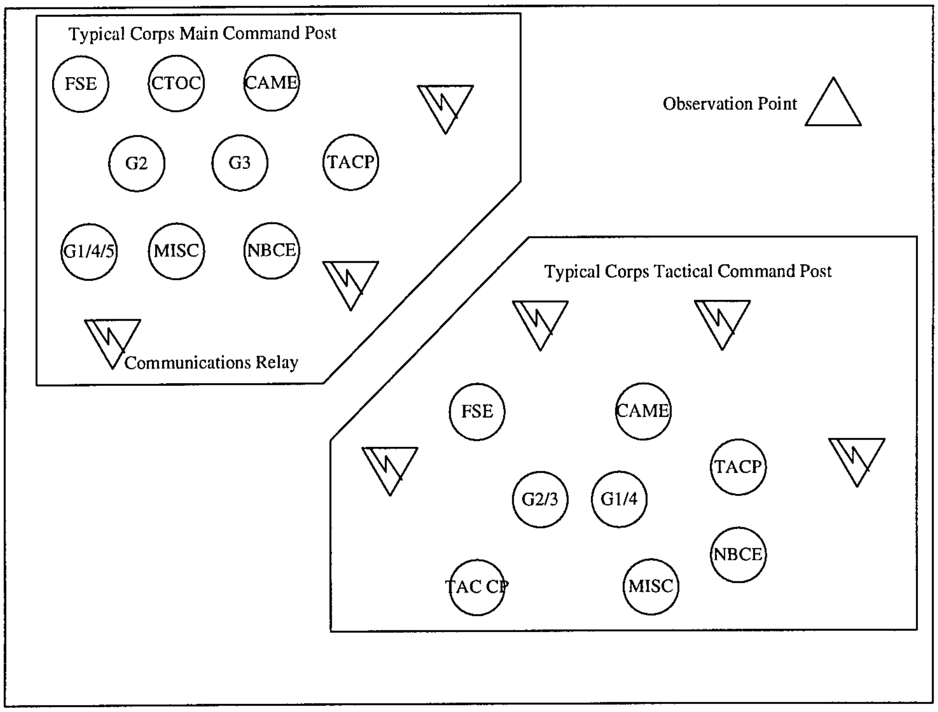


Figure 3.1 Typical Corps Command Post Composition. (Modified from [Dep84, Dep85]).

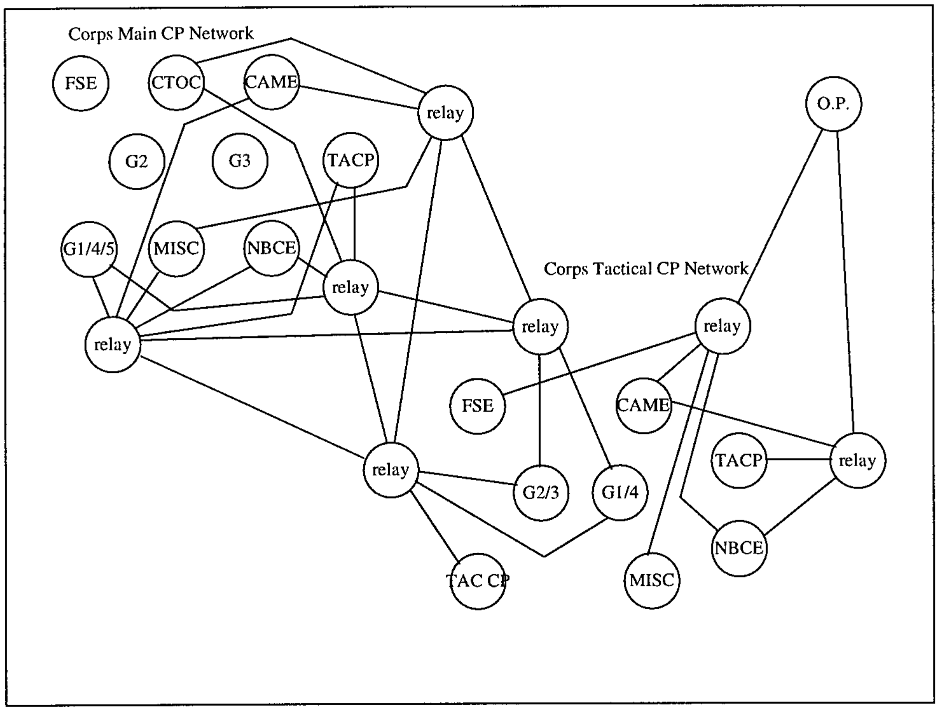


Figure 3.2a 25 node network topology with external relay links shown (based upon Figure 3.1)

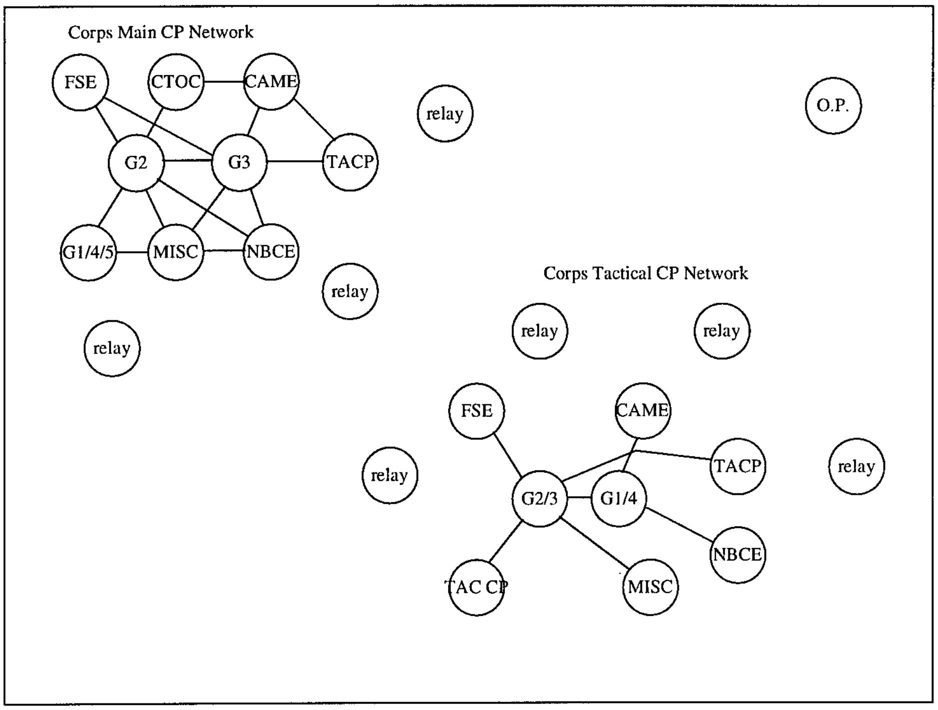


Figure 3.2b 25 node network topology with internal links shown (based upon Figure 3.1)

***3.3.2 Customer.*** The user defines the end to end system characteristics of the OSI transport layer. Packets entering a Packet Switching Network (PSN) are first presented to an entry node by an attached host. Generally, each attached host within GNA represents a single class of customers. In this study, there are two classes of customers. A primary customer and many secondary customers.

***3.3.2.1 Primary.*** The user defines the primary customer. The user indicates the start node attached to the originating system or host. Typically, a packet needs to be delivered to another host attached at an exit node. In GNA, the user indicates the destination node attached to the terminating system or host. GNA represents the exit node's attached host in the same manner as the entry node's host.

On a fluid battlefield, LOCs may change orientation rapidly. Unexpected threats may require the rapid redeployment of combat forces to block or counterattack the enemy.

Unexpected opportunities may develop and disappear unless quickly exploited [Dep93]. The dynamics of the modern battlefield -- speed, complexity, and lethality -- require the very highest level of organization and operational efficiency with command posts at all levels of command. Automated and/or manual information systems must minimize the time required for administrative processing of information [Dep84]. The designated start and destination nodes therefore represent a rear site broadcasting an important message to a forward site. This important message has a higher priority than other traffic in the network. Since the originating host at the start node is aware of the higher priority of the message, this importance can be represented with a higher traffic rate than the remaining routine traffic in the system.

The primary customer's arrival rate parameter, or Primary Traffic Rate (PTR), representing the higher priority message is defined by the user. This study uses the standard Poisson arrival process. The choice for the Poisson arrival process for use in QNA is explored in more detail in the QNA section [Whi83b].

In a typical switching node in a PSN, packets enter and leave the node by a set of incoming and outgoing links. As packets enter the node from one link from any number of possible incoming links, they are examined by the node's server. Based on this examination, the packet is scheduled for transmission on the appropriate outgoing link. The node servers are represented by GNA as it initially places the primary customer's incoming packet onto the appropriate outgoing link based on a routing table. This routing table contains the shortest route from the packets' source node and destination node. In a connection-oriented service, the routing table with the shortest path will be determined prior to the acceptance of the packet into the network.

***3.3.2.2 Secondary.*** The user also defines the traffic of the secondary customers. The user defines the Basic Secondary Traffic Rate (BSTR) for the entire network. This is the amount of traffic a node expects to service for each link attached to the node. The choice of using this BSTR method is evident after examining options below.

The BSTR is a factor that determines the amount of traffic to be encountered at each node. To represent the amount of traffic at each node within the network, QNA calculates the Total Sojourn Time at each node. This Total Sojourn Time is defined as the sum of Total Waiting Time at the node queue and the Total Service Time at the server.

Three options were considered in representing the amount of traffic at each node:

Option 1. The user defines the traffic rate expected to arrive at each node within the topology.

Option 2. The BSTR represents each link coming into the node where they are merged additively.

Option 3. The BSTR represents each link coming into the node and they are merged using Whitt's QNA procedures [Whi83b].

Option 1 ideally allows individual identification of every node's traffic within the network to give a accurate representation of the entire system's load. Unfortunately, it is not a real expectation for the user to have access to accurate traffic destinations of every possible start and destination node combination. Therefore, pursuing this option would yield no more accurate detail.

Option 2 allows the representation of a node with more links to receive more traffic. An example would be an important centralized and coordinating node whose function is to gather information from outlying nodes. The higher number of links is characteristic of such a node. The summed BSTR values based on the number of links would appropriately increase the traffic the node should expect to receive. A disadvantage of using option B's additive nature is the possibility the final traffic rate into a node may be greater than the capability of the node processing. The effects of this are shown in a later Section 3.3.3 regarding Network Flow Control.

Option 3 shares the same representation advantages of option 2. It uses a more robust procedure involving Whitt's superposition technique for the variability parameters developed for QNA. The QNA section reviews this technique in detail. Although the apparent disadvantages of option 3 appear like those of option 2, the QNA techniques counter its effect. Therefore option 3 most accurately represent the secondary traffic conditions within the GNA and QNA environments.

***3.3.3 Network Congestion Control.*** GNA emulates some of the characteristics of both the OSI model's transport layer and the network layer. GNA attempts to estimate the viability of the traffic within the network to determine an elementary form of Network Congestion Control based on the GNA Quality of Service (QoS). The basic GNA QoS is good if the nodes do not become unstable given the user-defined BSTR value and node service time parameters. Node instability is determined within QNA. This occurs when a node's traffic intensity or utilization is equal or greater than one [Whi83b]. When a node is unstable the basic GNA QoS is not satisfied. The GNA Network Congestion Control rejects the BSTR value and requests the user to redefine another BSTR value for the given current network topology.

GNA displays the rejection notification and informs the user certain network input parameters must be lowered (PTR or BSTR) or where certain nodes must be improved to maintain stability. Increasing either the number of servers or the processing capability of the nodes GNA identifies can relieve the instability. Once this instability is removed, a good QoS is achieved.

### *3.4 Route Determination*

There are a number of route determination considerations in this study, such as types of routing service, which was reviewed in Chapter Two. Other considerations are approaches to the routing function implementation, classification of the routing procedure, and the main components of a customer's packet sojourn time through a typical node.

Once the basic GNA QoS is satisfied, a routing service is required to determine the route from the source node to the destination node. There are generally two types of routing services that a packet may use. The first is connectionless service, which does not guarantee reliable delivery of messages [SaA94]. In a military environment, information must reach commanders in a timely manner for a decision to be made [Dep93]. Therefore connectionless service is not used in this study. GNA implements the second type of routing service, connection-oriented service. In connection-oriented service, a connection is established between the source and destination hosts before data is transferred. The connection also needs to be explicitly torn down at the end of data transfer. Regardless, all data appears to be carried over an end-to-end logical pipe.

The subnetwork within GNA provides the connection-oriented service, which is also known as external virtual circuit. With virtual circuit routing, GNA sets up a connection between the entry and exit nodes. As part of the connection set up procedure, GNA selects a path (virtual circuit) through the network. This path is followed by all packets traveling from entry node to exit node along the nodes belonging to the virtual circuit. Since all packets follow the same path, reliable and sequenced arrival of packets to the exit node is guaranteed.

GNA maintains the route used by the packet. The two basic approaches to implementing the routing function are Table-driven routing and Table-free routing. Since each node requires some degree of processing or service of the customer's packet to represent the examination of the destination address, this study uses the Table-driven approach. Given that the choice for virtual circuit service is used, a table with a listed route is maintained within GNA as with the Table-driven routing approach. Although a separate table is normally kept at each node of the network and updated constantly, this study involves a single user-defined source node. This allows GNA, which represents the network above the data link layer, to keep only a single table to conserve processing time and storage space. This will be discussed further in the Dijkstra's Algorithm portion of this section.

GNA emulates the network layer's path determination procedure. The inputs and outputs of path determination follow a centralized routing procedure. The inputs are provided to the centralized GNA for calculation. GNA simulates the user's network topology and adapts a modified path discovery method with a "single-route broadcast" type of forwarding. With the use of this type of forwarding, the path discovery occurs as a single-route broadcast. This single-route broadcast is sent from the user-defined source node on the simulated network. This path discovery is started within GNA by applying Dijkstra's algorithm upon the simulated network. As the broadcast reaches the user-defined destination node, a route is generated based on the destination. The generated route is kept in a local cache of the GNA data structure. GNA then outputs the generated route to QNA for use when it is needed. Although the path may provide minimum delay when chosen, there is no guarantee the path will continue to provide a minimum delay.

A customer's packet delay through a node has three major components: queueing time in the node server, processing time of the server, and packet transmission time. The transmission time is a function of the packet length and the data rate of the outgoing link. Since this study is dealing with a single form of traffic, the packet lengths are constant throughout. Therefore the transmission time is only a function of the data rate of the outgoing link. When the rate is low, the transmission time dominates the server processing time. But the data rate of the outgoing link modeled within GNA operates with the same assumptions mentioned earlier. That is, in modern packet switching networks, fiber optics links are used to accommodate fast data transmissions. This reduces a packet's transmission time by orders of magnitude, making the server processing time a more significant component of a packet's delay [SaA94]. Consequently, the packet's transmission time is not used.

***3.4.1 Dijkstra's Algorithm.*** The use of the Dijkstra's algorithm by GNA will guarantee a path between the source and destination nodes with the lowest total of "Sojourn Time". Although the user defines a single end node within GNA, the Shortest Backward Path Tree algorithm could have been more appropriate in this situation. Yet Dijkstra's algorithm take less time to compute than the Shortest Backward Path Tree algorithm [SaA94].

Given below is the plain language form of the general algorithm used (Figure 3.3). The two letter code preceding each instruction are the line numbers given for reference.

Dijkstra's algorithm starts with a set *N* initially containing all the nodes within the network. The algorithm maintains a set *S* whose *Current Traffic Value* from the source node has already been determined to be the lowest possible. It repeatedly selects a node *u* from the set *N-S* with the lowest *Current Traffic Value*, inserts node *u* into set *S*, and relaxes all edges leaving from node *u*. The priority Queue contains all the nodes in set *N-S*, keyed by the *Current Traffic Value*. The Adjacency Matrix contains the existing linkages between the nodes. The Traffic Graph contains the mean sojourn times expected at each node. The Predecessor Node array contains the node's predecessor as an element of the set *S*.

```
    DIJKSTRA (N, Adjacency_Matrix, Traffic_Graph, source_node)
D1  Call Initialization (N, source_node)
D2  S <- nil
D3  Queue <- Current_Traffic_Value(N)
D4  while Queue is not empty
D5    do  working_node <- Extract_Min (Queue)
D6        S <- S union {working_node}
D7        for each adjacent_node $\in$ Adjacency_Matrix[working_node]
             do Relaxation (working_node, adjacent_node, Traffic_Graph, Current_Traffic_Value)

    INITIALIZATION (N, source_node)
I1  For each node $\in$ N
I2    do    Current_Traffic_Value[node] <- Max_value
I3        Predecessor_Node[node] <- Nil
I4  Current_Traffic_Value[source_node] <- 0

    EXTRACT_MIN (Queue)
E1  Sort the Queue of nodes with the highest to lowest according to the Current_Traffic_Value
E2  Removes from the Queue and returns the node with the lowest Current_Traffic_Value

    RELAXATION (working_node, adjacent_node, Traffic_Graph, Current_Traffic_Value)
R1  u=working_node
R2  v=adjacent_node
R3  if Current_Traffic_Value[v] > Current_Traffic_Value[u] + Traffic_Graph (u, v) then
R4        Current_Traffic_Value[v] <- Current_Traffic_Value[u] + Traffic_Graph (u, v)
R5        Predecessor_Node[v] <- u
```

Figure 3.3 Plain language form of Dijkstra's algorithm used in the GNA tool.

Once all nodes all removed from the Priority Queue, the Predecessor Node Array is used to construct a route from the source node to the end node with the lowest possible traffic value. Although not formally considered a part of Dijkstra's algorithm, the generated route is a sequence of nodes from source to end with the total lowest possible traffic values or "Sojourn Times." This generated route will be distributed to QNA when it is needed. The full correctness proof can be found in [CoL90].

### *3.5 Generalized Network Analyzer Routing Algorithm*

This section displays two sample networks which verifies and empirically validates the routing algorithm within the GNA tool.

***3.5.1 GNA Chooses A Complete Path.*** Figure 3.4 shows Network A, a simplistic network, where the start node is chosen to be node 1 and the end node is chosen as node 3. The goal of Network A is to illustrate if the algorithm actually chooses a complete path from the start node to the end node. Given this network topology, the only option possible as a path is nodes 1, 2, and 3. Therefore, the algorithm output must chose the path composed of nodes 1, 2, and 3. Using a BSTR value of 3 and a PTR value of 5, the tool is executed. Upon completion of this experiment, the text output does indicate the path "1 2 3" as the sequence of nodes upon the chosen path. The Expected Sojourn Time is 1.51950. Appendix A contains this output.

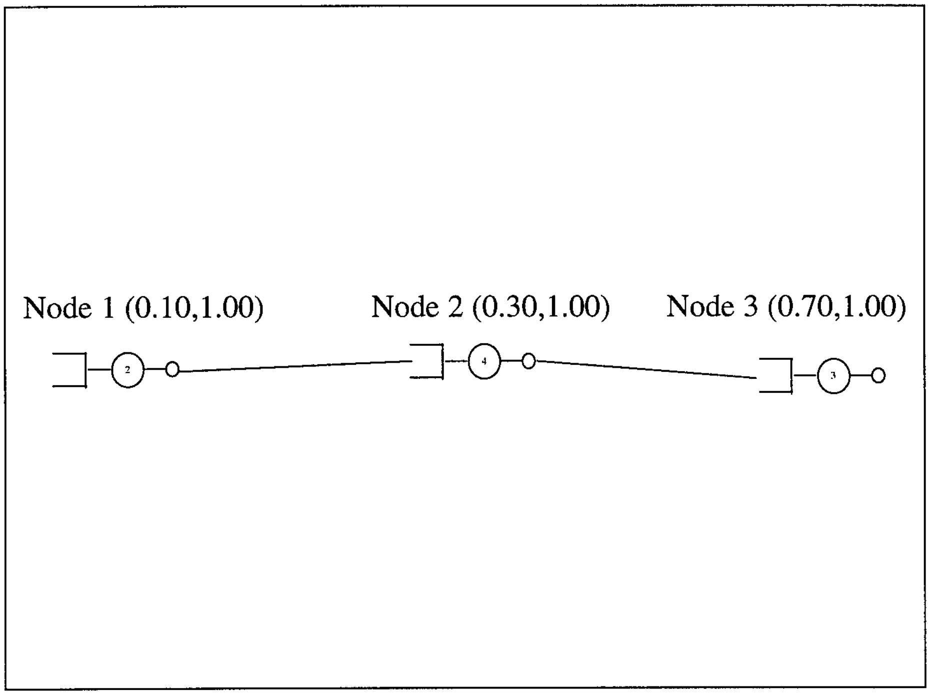


Figure 3.4 Network A shows that GNA chooses a complete path.

Network A serves to verify the algorithm will choose a path from beginning to completion. Another sample network is required to show the algorithm can actually determine the path with the lowest traffic congestion.

***3.5.2 Sojourn Times with the Standard Dijkstra's Algorithm.*** The goal of Network B, as shown in Figure 3.5, illustrates if the GNA algorithm actually chooses the least congested path from start node to end node. This is accomplished by calculating the Total Sojourn Time after using the standard Dijkstra's algorithm. To simplify matters, the topology was set up to allow the possible routes to be:

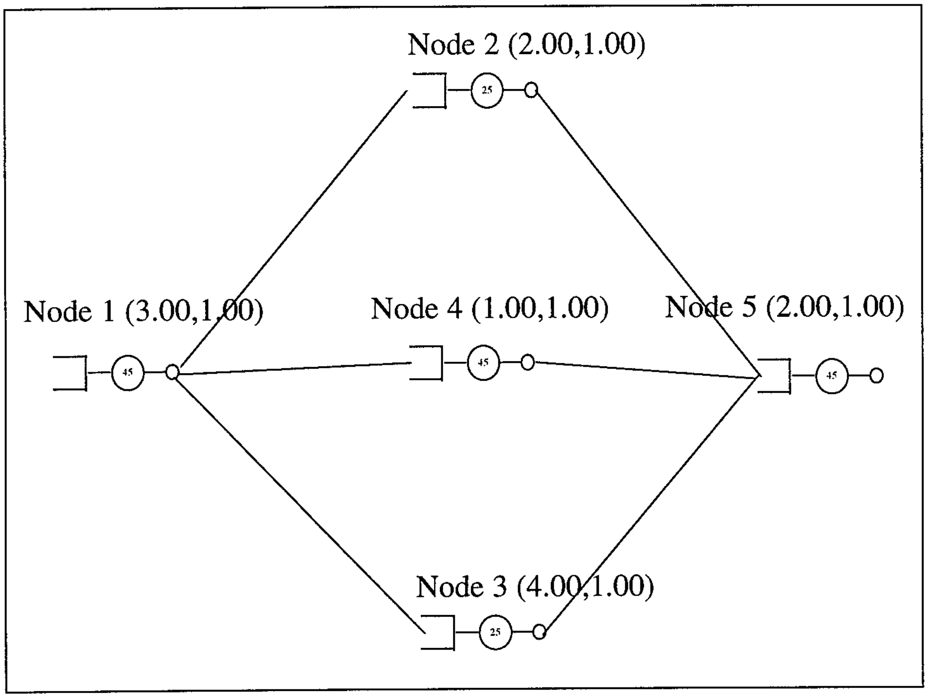


Figure 3.5 Network B shows that GNA chooses the least congested path.

Route "A" to be nodes 1-2-5,

Route "B" to be nodes 1-3-5, or

Route "C" to be nodes 1-4-5.

In this case, the total sojourn time is found by summing the individual sojourn times at each of the route's nodes (where *ST(1)* = sojourn time at node 1 and *ST(1)* + *ST(2)* + *ST(5)* = *ST(A)* is the sum of the individual sojourn times.)

$$ST(A) = ST(1) + ST(2) + ST(5) \tag{3.1a}$$

$$ST(B) = ST(1) + ST(3) + ST(5) \tag{3.1b}$$

$$ST(C) = ST(1) + ST(4) + ST(5) \tag{3.1c}$$

The standard sojourn time formula for an M/M/m queue [Jai91] is

$$ST = \tau\left[\frac{1+Q}{m(1-\rho)}\right] \tag{3.2}$$

where $Q$ is defined as the probability of queueing

$$Q = \left[\frac{(m\rho)^m}{m!(1-\rho)}\right]P_0 \tag{3.3}$$

and the symbol $P_0$ represents the probability of zero customers in the system. $P_0$ is given as [Jai91]

$$P_0 = \left\{1 + \left[\frac{(m\rho)^m}{m!(1-\rho)}\right] + \sum_{n=1}^{m-1}\left[\frac{(m\rho)^n}{n!}\right]\right\}^{-1} \tag{3.4}$$

The total delay values for each of the three routes using Equation (3.4) are listed in Table 3.1. It shows that Route C has the lowest Total Sojourn Time. Appendix B.1 contains all the arrival rate parameters the calculations done using Microsoft Excel.

Table 3.1 Total Sojourn Times of the three possible routes using standard queueing formula [Jai91].

| Routes | Nodes | Total Sojourn Time |
|---|---|---|
| A | 1-2-5 | 6.51826E+00 |
| B | 1-3-5 | 7.03412E+00 |
| C | 1-4-5 | 6.00294E+00 |

To validate the GNA methodology, the GNA algorithm must indicate the same route as the path determined by the standard Dijkstra's algorithm.

***3.5.3 Sojourn Times with the GNA Routing Algorithm.*** This section explains that the GNA methodology determines the same least congested path and final total sojourn times as the standard Dijkstra's algorithm would determine. A method for determining the shortest path is Dijkstra's algorithm. The weights within Dijkstra's algorithm are the costs involved for using a specific arc between two nodes of a graph. As explained earlier, the methodology used in this study is modified from the standard Dijkstra's algorithm. QNA calculates the congestion values (the goodness metric) at each node based on the BSTR value and the number of links associated with the node. The congestion value is the measurement of traffic that is expected at the node. The node's value is the cost associated with using the link destined for that node. The GNA algorithm determines the least congested path based on these values. Once the least congested path is determined, the PTR value will define the amount of traffic from the start node destined for the end node. Based on both the BSTR and the PTR values, QNA determines the total expected sojourn time from start to the end node using the least congested path.

Of the three routes within Network B, the one with the lowest total sojourn time will be determined by the routing algorithm to be the best path to use. With all possible routes, nodes 1 and 5 are common to all three routes. This is not surprising since node 1 is the start node and node 5 is the destination node as defined by the user. Therefore, the sojourn times of node 1 and node 5 can be removed from the total of each route and not alter which route has the lowest total sojourn time:

$$ST(A) = ST(2) \tag{3.5a}$$

$$ST(B) = ST(3) \tag{3.5b}$$

$$ST(C) = ST(4) \tag{3.5c}$$

It should be noted that the sojourn times used during the calculation within GNA do not remove any nodes. It is shown here to simplify calculations.

The sojourn time within GNA is simply the sum of all the nodes' waiting times before service and all the nodes' service time. Using some QNA notation (completely described in the QNA section), the value of

$$ST(k) = \sum_{j=1}^{n_k} E[W_{n_{kj}}] + \sum_{j=1}^{n_k} \tau_{kj} \tag{3.6}$$

where $n$ is the $n$ th node in route k (A, B, or C). Since $ST(k)$ is reduced to one node for calculations, the summations subscripts can be dropped to give

$$ST(k) = E[W_k] + \tau_k \tag{3.7}$$

where $E[W_k]$ is the expected waiting time and $\tau$ is the service time at the associated node. In order to use same arrival rate parameters as Appendix B1, a BSTR value of 2 is used. The QNA output values for $E[W]$ and $\tau$ are listed below in Table 3.2 for nodes 2, 3, and 4 along with their sojourn times. Appendix B.2 contains the entire QNA congestion calculation output for Network B.

Table 3.2 shows node 4 with the lowest sojourn time of the three nodes. Therefore, the GNA algorithm determines that Route C (nodes 1,4, 5) is the path with the lowest congestion.

Table 3.2 QNA approximated congestion outputs for selected nodes in Network B.

| Route / Node ID | Waiting Time (E[W]) | Service Time (τ) | Sojourn Time (ST) |
|---|---|---|---|
| A / 2 | 0.44147-04 | 0.20000+01 | 0.20000+01 |
| B / 3 | 0.11318-01 | 0.40000+01 | 0.40113+01 |
| C / 4 | 0.00000+00 | 0.10000+01 | 0.10000+01 |

After GNA determines the route with the lowest possible traffic value, GNA activates QNA to accomplish the queueing portion of the analysis.

***3.5.4 Expected Sojourn Time on Generated Path*** The final sojourn time on the chosen route is determined using the same formula as Equation (3.6). The QNA output, given the BSTR value of 2 and the PTR value of 3, for the chosen route is listed in Table 3.3. The sojourn time

Table 3.3 QNA outputs for expected waiting, service, and sojourn time of the nodes within the chosen route of Network B.

| Node ID | Waiting Time (E[W]) | Service Time (τ) | Sojourn Time (ST) |
|---|---|---|---|
| 1 | 0.29352-02 | 0.30000+01 | 0.30029+01 |
| 4 | 0.00000+00 | 0.10000+01 | 0.10000+01 |
| 5 | 0.49725-06 | 0.20000+01 | 0.20000+01 |
| Route C | 0.29352-02 | 0.60000+01 | 0.60029+01 |

for the chosen route using the GNA algorithm is 6.0029 units. Appendix B.3 contains the entire QNA sojourn time calculation for Network B.

Both the standard Dijkstra's calculations and the GNA algorithm indicated Route C as the route with the least congested path. The Total Sojourn Times of both method were identical to 5 significant digits. Network B empirically validates the GNA methodology. The next section covers the heart of the QNA model developed by Whitt.

### *3.6 Queueing Network Analyzer (QNA)*

***3.6.1 Introduction.*** The Queueing Network Analyzer (QNA) was developed to calculate approximate congestion measures for a network of queues. QNA can accommodate other models that do not use a Poisson arrival process nor exponential service-time distribution. QNA accomplishes this by characterizing each arrival process and each service-time distribution with a variability parameter. But for this study, QNA uses the Poisson arrival process and an exponential service-time distribution. The use of the Poisson arrival process and exponential service-time distributions allow for exact calculations of the queueing and service times.

The following is a quick description of the model. The network is represented by a set of nodes and directed arcs. The nodes represent the service sites and the arcs represent flows of customers, in our case primary and secondary customers. There is an additional external node that is not a service site but represents the outside world. From this external node the customers enter the network to an internal node. The customers travel from internal node to internal node by the directed arcs. It leaves the network by another external node. The customers' flow on the arcs are assumed to be random so they can be represented as stochastic processes. When the customer arrives and all the servers are busy at the node, the customer joins a queue and waits until the server is free. Service times at each node are assumed to be random variables. They are independent of the history of the network and are mutually independent and identically distributed. Customers flow from node *i* to node *j* and from node *j* to node *i*. QNA is set up so the customer may or may not be allowed to visit the same service site. Using the modified Dijkstra's algorithm, GNA will not generate a route that will reuse a node twice.

***3.6.1.1 Basic Assumptions.*** QNA is used to calculate approximate congestion measures. The use of QNA within GNA for this study is not to create a simulation system. QNA cannot represent successive time intervals to generate distinct traffic messages. Specific individual messages cannot be identified for modification. In the same manner, GNA cannot designate QNA to process a customer's route "A" at one interval and then process the same customer to route "B" in the next interval. Therefore, this packet switch study will not perform dynamic routing but concentrate on the virtual circuit aspect.

The QNA algorithms are used to approximate congestion in a long term steady state condition once all nodes within the network are stable. The flexibility of QNA allows specific identification of customer classes with certain parameters rather than specific messages.

There are 6 basic assumptions within QNA.

Assumption 1. The network is open rather than closed. Customers come from the outside, receive service at one or more nodes, and eventually leave the network.

Assumption 2. There are no capacity constraints. There is no limit on the number of customers that can be in the network and each service site has infinite buffers.

Assumption 3. There can be any number of servers at each node. They are identical independent servers, each serving one customer at a time.

Assumption 4. Customers are selected for service at each site according to the first come, first serve discipline.

Assumption 5. There can be any number of customer classes, but customers cannot change classes. Much of the analysis in QNA is done for the aggregate or typical customer. This study examines two customer classes.

Assumption 6. Customers can be created or combined at the nodes. An arrival can cause more than one departure. This study will not require any creation or combination of customers.

QNA's general approach represents all the arrival processes and service-time distributions with just a few parameters. These parameters are also used in approximate formulas that describe the congestion at each service site. The parameters for the internal flows are determined by applying elementary calculus on the parameters to represent each of the three basic network operations: merging (superposition), splitting (thinning), and departure (flow through a queue). When the queues are in a series, the basic calculus can be applied successively one at a time. It is sometimes necessary to solve a system of equations. There are four key elements in QNA's approach to represent all the arrival processes and service-time distributions:

1. Parameters characterizing the flows and nodes will be readily available in the GNA application and have considerable descriptive power in approximations of the congestion at each node.

2. Approximations for multi-server queues are based on the partial information provided by the parameters characterizing the arrival process and the service-time distribution at each node.

3. A calculus for modifying the parameters that represent the basic network operations: merging, splitting, and departure.

4. A synthesis algorithm to solve the system of equations resulting from the basic calculus applied to the network.

QNA uses two parameters to characterize the arrival process and the service times, one to describe the rate and the other to describes the variability. For the service times, the two parameters are the first two moments. QNA actually uses the mean service time and the squared coefficient of variation. The squared coefficient of variation is the variance of the service time divided by the square of its mean. For the arrival process, the two parameters are equivalent to the first two moments of the renewal interval in the approximating renewal process. QNA uses the arrival rate and the squared coefficient of variance. The squared coefficient of variation is the variance of the renewal interval divided by the square of its mean.

QNA uses an approximation method to obtain the flows into the queues. QNA starts with Whitt's basic procedure of approximating point processes [Whi82] plus refinements from other authors [Whi83b], which will be described in later sections. Whitt references Kuehn to describe the approximation method in QNA as a parametric-decomposition method. Parametric-decomposition is when the nodes are analyzed separately after the parameters for the internal flows are determined [Kue79]. It is important to note that QNA approximates the nodes as being stochastically independent in order to calculate the congestion for the network as a whole. This is interpreted as a generalization of the product-form solution in a Markovian network. Even though QNA is described as a decomposition method or an extended product-form solution, Whitt also attempts to represent the dependency among the nodes. QNA accomplishes this dependency by using internal flow parameters, which will be discussed more in later sections.

To illustrate QNA, consider a simple open network with a single node, a single server, infinite buffer space, and a first-come, first-served discipline (Figure 3.6).

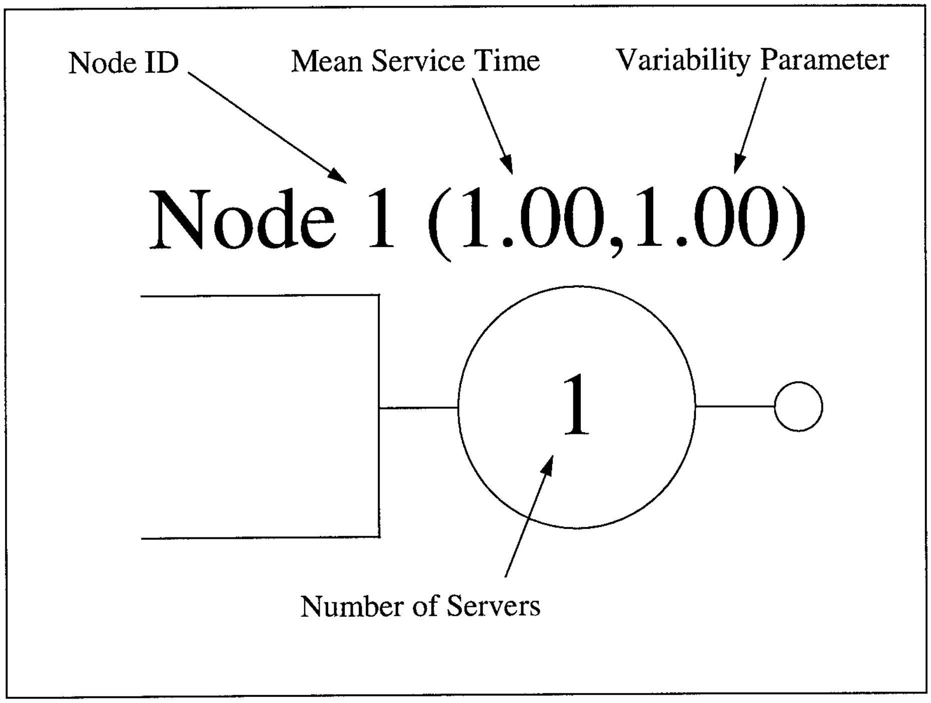


Figure 3.6 Simple M/M/1 Queue in GNA format for use in QNA.

Consider a single class of customers that is served once before departing the network. The standard Markov model is the classical M/M/1 queue. For this M/M/1 model, the expected waiting time (EW) is

$$EW = \frac{\tau\rho}{(1-\rho)} \tag{3.8}$$

where $\tau$ is the mean service time and $\rho$ is the traffic intensity, where $0 \leq \rho < 1$.

Within QNA, the one node network is approximated as a GI/G/1 model. The GI/G/1 model has a renewal arrival process and both the interarrival-time distribution and the service-time distribution are general. The arrival process is represented by a renewal process partially characterized by two parameters. The arrival rate is $\lambda$ and the variability parameter is $c_a^2$ .

The service-time distribution is also partially represented by two parameters. The mean service time is $\tau$ and the variability parameter is $c_a^2$ . The expected waiting time, $(EW)$ , is

$$EW = \frac{\tau\rho(c_a^2 + c_s^2)g}{2(1-\rho)} \tag{3.9}$$

where $g \equiv g\ (\rho,\ c_a^2\ ,\ c_s^2)$ is either one, when $c_a^2 \geq 1$ , or less than one, when $c_a^2 < 1$ . When $g(\rho,\ c_a^2,\ c_s^2)= 1$ , Equation (3.9) differs from Equation (3.8) by the factor $\frac{\left(c_a^2 + c_s^2\right)}{2}$ . When the arrival process is Poisson, $c_a^2 = 1$ ; when the service-time distribution is exponential, $c_s^2 = 1$ . Therefore, QNA's GI/G/1 model is seen as an M/M/1 model. The default values for $c_a^2$ and $c_s^2$ in QNA are one. This represents the primary and secondary customer traffic in this study closely. Whitt points out if there is a single class of customers, if all the arrivals are Poisson, and if all the service-time distributions are exponential, then QNA is exact [Whi83b]. This study will include a separate customer class for each node to represent the arrival of traffic to the node. Whitt's approximations to determine the internal flow parameters is used.

The next few paragraphs describe the workings of QNA in three parts, input, internal flows, and output. The sections following give further details.

***3.6.1.2 Input.*** The user of QNA has a number of input options. QNA has two input options, the standard input by parameters and the alternate input by classes and routes.

The standard input method only requires a limited amount of information. Two parameters are needed for each service-time distribution and two parameters are needed for each external arrival process. A routing matrix is also used to describe the probability of customers traveling from a specific node to another specific node.

This study uses characteristics of the alternate input method. The alternate input method occurs when each different class of customers enters the network at a fixed node and passes through a specified sequence of nodes. For each class, there are two parameters to describe the external arrival process and two parameters to describe the service-time distribution at each node on its route. With this method, each class can have a different service-time distribution at a given node and different service-time distribution during different visits to the same node. Since in this study, the primary customer is not distinguished until it is processed, each node treats all the classes with the same service rate. It is important to note QNA analyzes this route input by aggregation. All classes are aggregated by QNA to convert the route input into the standard input method. Afterwards, the special parameters of each class are used to describe its total travel time.

***3.6.1.3 Internal Flow.*** QNA determines the internal flow within each of the nodes. QNA uses calculus to transform the arrival parameters to characterize the internal flow. Since the calculus is linear for each network operation, the parameters are determined by solving a system of linear equations. The internal flow arrival rate parameters are just the traffic rate equations of a Jackson network of M/M/m queues. The internal flow variability parameters (the squared coefficient of variations) are determined by solving another system of linear equations.

***3.6.1.4 Output.*** QNA outputs a number of system and customer measures. For each node, QNA outputs the traffic intensity, the expected number of busy servers, the mean and variance of the equilibrium delay and the number of customers present. As indicated earlier, congestion measures for the entire network are given under the assumption that the nodes are stochastically independent given the approximate internal flow parameters. For the customers, means and variances of total service times, total delays, and total travel times are calculated.

When the standard input is used, these values are given by routes. When the alternate input is used, the values are given for each customer class.

The remaining sections describe QNA in even more detail. The Input section describes in detail the required data variables for both the standard method and the alternate method. The Internal Flow Parameters section describes in detail the traffic rate equations, the system of linear equations that display the variability parameters of the internal flow. The formation of the linear equations of the variability parameters are explained further by examining the basic operations of superposition, thinning, and departure. Synthesis will explain how the basic operations form the internal flow variability parameter equations. The Congestion at the Nodes section discusses how QNA treats the single-server node and the multi-server node. Lastly, the Total Network Performance Measures section describes the calculations on a system level and customer congestion measures from a particular customer's view of a specific route.

***3.6.2 Input.*** QNA accepts two versions of input. Both versions are described below. The network uses the second form of input.

***3.6.2.1 Standard Input.*** QNA works with a standard input. Although any number of networks can be processed during a single run, the current configuration of GNA shows it works with one network at a time [See84]. Within each network, the number of nodes is indicated, each with a specific number of servers. For each node, two parameters are used to describe the external arrival process and two parameters to describe the service-time distribution. There is also a routing matrix to indicate the proportion of customers that go to node $j$ from node $i$ . Of the GNA symbols listed from Section 2.2.1, Figure 3.7 shows a list of the input notation used specifically within QNA.

$n$ = number of internal nodes in the network
$m_j$ = number of servers at node j
$\lambda_{0j}$ = external arrival rate to node j
$c_{0j}^2$ = external arrival variability parameter to node j
(squared coefficient of variation of the renewal interval in the approximating renewal process)
$\tau_j$ = mean service time at node j
$c_{sj}^2$ = squared coefficient of variation of the mean service-time distribution at node j
$q_{ij}$ = proportion of those customers completing service at node i that goes next to node j
In matrix notation, $Q = (q_{ij})$ is an n x n matrix and $\Lambda_0 = (\lambda_{0j})$ is an 1 x n vector.

Figure 3.7 QNA standard input notation.

The variability parameters, $c_{0j}^2$ and $c_{sj}^2$ , within QNA have a default value of one for this study. This corresponds to the M/M/1 model having a Poisson arrival process and an exponential service-time distribution.

The option within QNA to create or combine customers after the completion of service at each node is not a goal of this study. This option gives the capability to split one customer into identical customers to be sent to other nodes. This should be considered in future studies of this topic.

***3.6.2.2 Alternate Input.*** The method of input used in this study allows defining different customer classes. Each class has its own route that specifies the sequence of nodes. Each class has an external arrival process that goes to the first node in the sequence. Each class has its external arrival rate and variability parameters. QNA is flexible enough to allow each class to have a service-time distribution and variability parameters for each node it visits. These parameters are independent for each class and therefore can be different or unique. Within this study, the service-time distribution for a particular node is the same for all classes that visit that node. The number of nodes and servers at each node are also used in this method. In addition to Figure 3.7, Figure 3.8 shows a list of the input notation used with QNA.

This method requires additional input data.
$r$ = number of routes
$n_k$ = number of nodes on the route used by customer class k

Additional input data for the kth customer class normally required, but not needed here.
$n_{kj}$ = the jth node visited by customer class k
$\tau_{kj}$ = mean service time at the jth node on the route used by customer class k
$c_{skj}^2$ = squared coefficient of variation of the mean service-time distribution at the jth node on the route used by customer class k

Figure 3.8 QNA alternate input notation.

***3.6.2.3 Alternate Input Conversion.*** This method's input data is converted by QNA into the standard input which was described earlier. In this conversion, it calculates the parameters of a typical or aggregate customer. When computing the delay times or response times for a particular customer class, QNA uses the service-time parameters of that particular customer class. QNA assumes as an approximation that each customer sees independent versions of the equilibrium distribution at each node. Therefore, it is important to note the waiting time before beginning service at each node is assumed to be the same for all classes and all visits.

This section explains how QNA converts the alternate input into the standard input. Let the symbol *1H* be the indicator function of the set $H$. This means *1H(x) = 1* if $x$ is a member of set $H$ and *1H(x) = 0* otherwise. The external arrive rate is obtained by

$$\lambda_{0j} = \sum_{k=1}^{r} \hat{\lambda}_k 1\{k: n_{k1} = j\} \tag{3.10}$$

This equation is broken down in detail for a better understanding. This means the external arrival rate at node $j$, given with the symbol " $\lambda_{0j}$ ," is the sum of all the routes arrival rates, symbol " $\Sigma_{k=1..r} \lambda_{kj}$ ," whose first node of that route $k$, symbol " $k{:}n_{k1}{:}$ ," is node $j$, symbol "$=j.$" Here $1H$ is used for the set $H = \{k{:}n_{k1} : =j\}$. The flow rate from node $i$ to node $j$ is

$$\lambda_{ij} = \sum_{k=1}^{r} \sum_{\ell=1}^{n_k - 1} \hat{\lambda}_k 1\{(k, \ell): n_{k,\ell} = i, n_{k,\ell+1} = j\} \tag{3.11}$$

and the flow from node $i$ out of the network is

$$\lambda_{i0} = \sum_{k=1}^{r} \hat{\lambda}_k 1\{k{:}n_{kn_k} = i\} \tag{3.12}$$

From Equations (3.11) and (3.12), the routing matrix $Q$ can be created. The proportion of customers going from node $i$ to node $j$ is

$$q_{ij} = \frac{\lambda_{ij}}{\lambda_{i0} + \sum_{k=1}^{n} \lambda_{jk}} \tag{3.13}$$

So if node $i$ is used within the network, then the denominator will always be positive. Within QNA, if a node is given but not used, a error message will be supplied to GNA. This study will ensure all nodes within the user defined network topology will have traffic.

As mentioned in an earlier section, this study assumes the service time for each server doesn't change between primary or secondary customers. The values of the service-time parameters and variability parameters are the same as the standard input service parameters.

Now QNA has enough standard input information to compute the internal flow rates $\lambda_j$ and the traffic intensities $\rho_j$ for use in the Internal Flow Parameter section:

$$\rho_j = \frac{\lambda_j \tau_j}{m_j} \tag{3.14}$$

QNA uses this information to calculate the variability parameter $c_{0j}^2$ of the external arrival process. Whitt uses superposition (merging) for the arrival process variability because the external arrival processes to node $j$ is the merging of the external arrival process to node $j$ from all the other involved classes. In this study, the involved classes are the links that end at each node and an additional class if the node is a member of the primary customer's route. Therefore a node along the primary customer's path with three links will be merged as a total of four classes. Likewise, a node (not the start or end nodes) with one link, and therefore not included in the primary customer's route, will only involve one external arrival class. The merging is actually a hybrid approximation Whitt developed [Whi83b] which he uses mainly to describe the variability parameters of the internal flow. This is found later in the Internal Flow Parameter section.

***3.6.2.4 Immediate Customer Feedback.*** Although Kuehn suggested eliminating immediate customer feedback as an option within QNA [Kue79], it is not necessary for this study since a network with an attached host will never send a message to itself. Normally, immediate feedback occurs when $q_{ii}>0$ . Since QNA is using Markovian routing, each customer completing service at node $i$ is fed back to node $i$ with probability $q_{ii}$. Each time the customer goes to the end of the line. As stated earlier, the customer will find an equilibrium number of customers at the node each time. Each time being an independent visit to the node.

What Kuehn suggested was to eliminate this feedback by placing the customer at the head of the line rather than at the end. This would seem appropriate as a server processes a message to discover itself as the next destination. It would not take that message and place it at the end of the queue, but would immediately continue processing it. Given this study, there is no requirement to send a message from a node to itself. A detailed explanation of eliminating immediate customer feedback can be found in Whitt's work on QNA [Whi83b].

***3.6.3 Internal Flow Parameter.*** There are a number of internal flow parameters for each node. Each node has traffic rate parameter, and a traffic rate variability parameter.

***3.6.3.1 Traffic Rate Equations.*** This step calculates the total arrival rate to each node. Let $\lambda_j$ be the total arrive rate to node $j$, let $\gamma_j$ be the multiplicative factor of customer creation at node $j$. Let $\delta_j$ be the departure rate at node $j$. In general, $\delta_j = \lambda_j \gamma_j$. In this study, there is no customer creation, $\gamma_j = 1$ and the rate in equals the rate out. Generally, the fundamental traffic-rate equation is

$$\lambda_j = \lambda_{0j} + \sum_{i=1}^{n} \lambda_i \gamma_i q_{ij} \tag{3.15}$$

for $j = 1, 2, ..., n$. In matrix notation

$$\Lambda = \Lambda_0 ( I - \Gamma Q )^{-1} \tag{3.16}$$

where $\Lambda_0 = (\lambda_{0j})$ is the external arrive-rate vector, $Q = (q_{ij})$ is the routing matrix, and $\Gamma=(\gamma_{ij})$ is the diagonal matrix with $\gamma_{ii} = \gamma_i$ and $\gamma_{ij} = 0$ for $i \neq j$. When there is no customer creation, $\gamma_i = 1$ and $\Gamma = I$. Equation (3.15) is just a system of linear equations. Solving them is equivalent to inverting the matrix $(I - \Gamma Q)$ in Equation (3.16).

With the arrival rates, the traffic intensities (utilizations) at each node are defined by

$$\rho_i = \frac{\lambda_i \tau_i}{m_i} \tag{3.17}$$

where $1 \leq i \leq n$. If $\rho_j \geq 1$, then the $i$ th node is unstable. If any node is unstable, QNA gives an error message, prints out the traffic intensities, and stops. This means a failure of the basic QNA QoS mentioned in the Network Flow Control section. The associated offered load at node $i$, which coincides with the expected number of busy servers is

$$\alpha_i = \lambda_i \tau_i \tag{3.18}$$

where $1 \leq i \leq \mathrm{n}$. The parameters $\alpha_i$ and $\rho_i$ coincide for a single server, with $\alpha_i$ being more applicable as the offered load when the number of servers, $m_i$, increases.

QNA calculates the following values for each arc: $\lambda_{ij}$, the arrival rate from node $i$ to node $j$ and $p_{ij}$, the proportion of arrivals from node $i$ to node $j$, when $i \geq 0$:

$$\lambda_{ij} = \lambda_i \gamma_i q_{ij} \tag{3.19a}$$

$$p_{ij} = \frac{\lambda_{ij}}{\lambda_j} \tag{3.19b}$$

QNA calculates the following output values: $d_i$ , the departure rate out of the network from node $i$ and $d$ represents the total departure rate out of the network.

$$d_i = \lambda_i \gamma_i \left( 1 - \sum_{j=1}^{n} q_{ij} \right) \tag{3.20a}$$

$$d = \sum_{i=1}^{n} d_i \tag{3.20b}$$

***3.6.3.2 Traffic Variability Equations.*** QNA's approximation is a system of equations that determines the variability parameters for the internal flows. The internal flow is the squared coefficient of variation for the arrival process, $c_{aj}^{2}$ . The linear equations are of the following form

$$c_{aj}^2 = a_j + \sum_{i=1}^{n} c_{ai}^2 b_{ij} \tag{3.21}$$

where $1 \leq j \leq n$ and $a_j$ and $b_{ij}$ are constants, where depending on the input:

$$a_j = 1 + w_j \{ [p_{0j} c_{0j}^2 - 1] + \sum_{i=1}^{n} p_{ij} [(1 - q_{ij}) + (1 - v_{ij}) \gamma_i q_{ij} \rho_i^2 x_i ] \} \tag{3.22}$$

and

$$b_{ij} = w_j \; p_{ij} \; q_{ij} \; \gamma_i \; [ \, v_{ij} + ( 1 - v_{ij} ) ( 1 - \rho_i^{2} ) \, ] \tag{3.23}$$

where $x_i$ , and $v_{ij}$, and $w_j$ depend on the basic data already provided. These values are

$$x_i \quad = 1 + m_i^{-0.5} (max \mid c_{si}^2 , 0.2 \mid - 1) \tag{3.24}$$

$$v_{ij} \quad = 0 \tag{3.25}$$

$$w_j \quad = [ 1 + 4 ( 1 - \rho_j )^2 ( v_j - 1 ) ]^{-1} \tag{3.26}$$

$$v_j = \left( \sum_{i=1}^{n} p_{ij}^2 \right)^{-1} \tag{3.27}$$

The parameter $\gamma_i$ is the same multiplicative factor of customer creation or combination as discussed earlier. The value of $\gamma_i$ is equal to one in this study. The variables $x_i$ and $v_{ij}$ are used to determine the departure operation. The variable $w_j$ is used to determine the superposition operation. The variables $v_{ij}$ and $w_j$ are weights or probabilities that are used in convex combinations in hybrid approximations for departure and superposition, respectively.

The next three sections goes in depth regarding how the variability parameters are determined for the internal flows. Whitt references his earlier work on asymptotic method and stationary-interval method [Whi82].

***3.6.3.2.1 Superposition.*** Using the asymptotic method, a linear method, the superposition squared coefficient of variation $c_A^2$ as a function of component squared coefficient of variation $c_i^2$ and the rates $\lambda_i$ is just the convex combination

$$c_A^2 = \sum_i \left( \frac{\lambda_i}{\sum_k \lambda_k} \right) c_i^2 \tag{3.28}$$

However, Whitt found that neither method alone worked very well over a wide range of cases [Whi83a]. Whitt references a refined composite procedure found by Albin [Whi83b], which is a convex combination of $c_A^2$ for the asymptotic method and $c_{SI}^2$ for the stationary interval method. The hybrid $c_H^2$ is of the form

$$c_H^2 = w\, c_A^2 + ( 1 - w )\, c_{SI}^2 \tag{3.29}$$

But since $c_{SI}^2$ is nonlinear, so is $c_H^2$ . Whitt used Albin's convex combination of $c_A^2$ and the exponential $c_{SI}^2 = 1$ worked well [Whi83b]. The new hybrid procedure is

$$c_H^2 = w\, c_A^2 + ( 1 - w ) \tag{3.30a}$$

$$= w \sum_i \left( \frac{\lambda_i}{\sum_k \lambda_k} \right) c_i^2 + 1 - w_i \tag{3.30b}$$

where $w$ is a function of $\rho$ and the rates. The Albin weighting function $w$ is

$$w = [ 1 + 2.1 ( 1 - \rho )^{1.8}\, v ]^{-1} \tag{3.31}$$

where

$$v = \left[ \sum_i \left( \frac{\lambda_i}{\sum_k \lambda_k} \right)^2 \right]^{-1} \tag{3.32}$$

But Whitt indicates if there are $k$ component processes with equal rates then $v = k$. The parameter $v$ can be thought of as the number of component streams, with it being an equivalent number if the rates are unequal. Whitt references his own unpublished theoretical results indicating the exponent of $(1\text{-}\rho)$ in Equation (3.31) should be equal to 2. Therefore, QNA uses the weighting function $w$ of Equation (3.26) for Equation (3.30b) instead of the weighing function $w$ of Equation (3.31).

***3.6.3.2.2 Splitting.*** Since a renewal process split according to a Markovian routing is a renewal process, the asymptotic method and the stationary-interval method coincide. When a stream with a parameter $c^2$ is split into $k$ stream, with each being selected independently with probabilities $p_i$, $i = 1, 2, ..., k$, then the $i$ th process obtained from the splitting has a squared coefficient of variation $c_i^2$ given by

$$c_i^2 = p_i \; c^2 + 1 - p_i \tag{3.33}$$

The renewal-interval distribution in the split stream is a geometrically distributed random sum of the original renewal intervals.

***3.6.3.2.3 Departures.*** Whitt explains that for the stationary-interval method with a single-server node, the squared coefficient of variation of interdeparture time, $c_d^2$, in a GI/G/1 queue using Marshall's formula is

$$c_d^2 = c_a^2 + 2\rho^2 \; c_s^2 - 2\rho \,( 1- \rho )\, \mu \, EW \tag{3.34}$$

where $EW$ is the mean waiting time. Therefore, the variability of the departure is affected by the congestion at the node. The approximation of $EW$ from Equation (3.9) is used with Equation (3.34) while $g$ in Equation (3.9) is set to the value of one will results in

$$c_d^2 = \rho^2 \, c_s^2 + ( 1 - \rho^2 ) \, c_a^2 \qquad (3.35)$$

QNA uses an extended version of Equation (3.35) for GI/G/m queues in the form of

$$c_d^2 = 1 + ( 1 - \rho^2 )( c_a^2 - 1 ) + ( \frac{\rho^2}{\sqrt{m}} )( c_s^2 - 1 ) \qquad (3.36)$$

Equation (3.36) describing a GI/G/m queue agrees with Equation (3.35) when $m = 1$. For M/M/m and M/G/∞ systems where the stationary departure process is known to be Poisson, the parameters, $c_a^2 = 1$, describes a Poisson process and, $c_s^2 = 1$, describes a exponential service-time distribution reduces $c_d^2$ to the value of one. As the number of servers $m$ increases in Equation (3.36), the departure process variability will depend more upon the arrival process variability. In cases when the service-time is deterministic, such that $c_s^2 = 0$, the departure process is less variable than the arrival process. Equation (3.35) or (3.36) under estimates the reduction in variability for deterministic service-times [Whi83b]. Therefore Whitt substituted (3.36) with a correction factor of $c_s^2 = 0.2$ when the original value of $c_s^2 = 0$

$$c_d^2 = 1 + ( 1 - \rho^2 )( c_a^2 - 1 ) + ( \frac{\rho^2}{\sqrt{m}} )( max\{ c_s^2 , 0.2 \} - 1 ) \qquad (3.37)$$

***3.6.3.2.4 Synthesis.*** The synthesis of the equations from the basic operations of superposition, thinning, and departure, Equations (3.21) through (3.27), results as follows:

$$c_{aj}^{2} = 1 - w_{j} + w_{j} \sum_{1=0}^{n} p_{ij} c_{ij}^{2} \tag{3.38a}$$

or when fully expanded

$$= 1 - w_{j} + w_{j} \left( \begin{array}{l} p_{0j} c_{0j}^{2} + \\ \displaystyle\sum_{i=0}^{n} p_{ij} \left\{ v_{ij} \left[ \begin{array}{l} q_{ij} \gamma_{i} c_{ai}^{2} + (1 - q_{ij}) \\ +(1 - v_{ij}) \left( \gamma_{i} q_{ij} \left\{ \begin{array}{l} 1 + \left[1 - \rho_{i}^{2}\right]\left[c_{ai-1}^{2}\right] \\ + \rho_{i}^{2} m_{i}^{-0.5} \left[\max(c_{SI}^{2}, 0.2)\right] \end{array} \right\} \right) + 1 - q_{ij} \end{array} \right] \right\} \end{array} \right) \tag{3.38b}$$

***3.6.4 Congestion At The Nodes.*** QNA has calculated the rate, $\lambda_j$ Equation (3.15), and variability parameters, $c_{aj}^2$ Equation (3.38a), for each internal arrival process at every node, the subscript $j$ representing the $j$ th node can be dropped. At this point, QNA can calculate the approximate congestion at each node. Each node is analyzed individually in isolation given five parameters, the number of servers $m$, the arrival rate $\lambda$, the mean service time $\tau$, and the squared coefficient of variation $c_a^2$ and $c_s^2$. Given that QNA only has five parameters, Whitt developed it to use the GI/G/1 queue since there is little to be gained by using more elaborate procedures [Whi83b]. In this study where the parameters, $c_a^2$ = 1, describing a Poisson process and, $c_s^2$ = 1, describing a exponential service-time distribution, the formulations given will be reduced to a simpler result. But a full explanation is given to show QNA's flexibility.

***3.6.4.1 GI/G/1 Queue.*** QNA uses the same mean waiting time, $EW$, formula as stated in Equation (3.9), it is repeated here.

$$EW = \frac{\tau\rho(c_a^2 + c_s^2)g}{2(1-\rho)} \tag{3.39}$$

where the value of $g$ is defined as one when the value of $c_a^2$ = 1, as it is in this study. This study will reduce (3.39) to

$$EW = \frac{\tau\rho}{(1-\rho)} \tag{3.40}$$

which is standard M/M/1 $EW$. Where $N$ is the number of customers in the node, the probability that a server is busy P[ $N$ > 0], and the expected number $EN$ can be found from Little's Formula as

$$P[N > 0] = \rho \tag{3.41}$$

and

$$EN = \rho + \lambda\, EW \tag{3.42}$$

The probability of delay (or waiting time) is greater than zero P[ $W$ > 0] given by the symbol $\sigma$ is

$$\sigma = P[W > 0] = \rho + (c_a^2 - 1)\,\rho\,(1 - \rho)\,h(\rho, c_a^2, c_s^2) \tag{3.43}$$

where $h(\rho, c_a^2, c_s^2)$ is what Whitt describes as the Kraemer and Langenbach-Belz approximation [Whi83b]. The symbol $h$, in this case where $c_a^2 = 1$ and $c_s^2 = 1$, reduces to

$$h(\rho) = \frac{4\rho}{(1+5\rho^2)} \tag{3.44}$$

The value of $\sigma$, in this study, will basically reduce to

$$\sigma = \rho \tag{3.45}$$

Whitt defines the delay, $D$, as when the server is busy and defines the conditional delay as

$$ED = \frac{EW}{\sigma} \tag{3.46}$$

Our study reduces to

$$ED = \frac{\tau}{(1-\rho)} \tag{3.47}$$

which is the standard expected delay of a M/M/1 queue. Equation (3.48) shown below is exact for an M/G/1 queue when the service-time distribution is hyperexponential, $H_2^b$. Whitt uses the approximation in Equation (3.48) because the conditional delay in a GI/G/1 queue (rather than total delay $W$ ) depends more on the service-time distribution than on the interarrival-time distribution [Whi83b]. Therefore, the approximation formula for the squared coefficient of variation of delay $D$ is

$$c_D^2 = 2\rho - 1 + \frac{4(1-\rho)d_a^3}{3(c_S^2+1)^2} \tag{3.48}$$

where $d_S^3 = \frac{E[v^3]}{(Ev)^3}$ with $v$ being a service-time random variable. But since $E[v^3]$ is not available with the study's two parameters QNA approximates $d_s^3$ based on the $H_2^b$ distribution,

$$d_s^3 = 3\ c_s^2\ (\ 1 + c_s^2\ ) \tag{3.49}$$

Once again, the squared coefficient of variation of delay $D$ is reduced to

$$c_D^2 = 1 \tag{3.50}$$

From Equations (3.39), (3.43), (3.48), and (3.49), the formula for $Var(D)$ and $E(D)^2$ are

$$Var(D) = (ED)^2 c_D^2 = \frac{(EW)^2 c_D^2}{\sigma^2} \tag{3.51}$$

and

$$E(D^2\ )= Var(D) + (ED)^2 \tag{3.52}$$

In this study, this is reducible to the standard M/M/1

$$Var(D) = \frac{\tau^2}{(1-\rho)^2} \tag{3.53}$$

Given $D$,

$$c_W^2 = \left(\frac{E(W^{2)}}{(EW)^2}\right) - 1 \tag{3.54a}$$

$$c_W^2 = \left(\frac{\sigma E(D)^2}{(\sigma ED)^2}\right) - 1 \tag{3.54b}$$

$$c_W^2 = \frac{c_D^2 + 1 - \sigma}{\sigma} \tag{3.54c}$$

and in reduced form

$$c_W^2 = \frac{(2-\rho)}{\rho} \tag{3.55}$$

the second-moment characteristics for $W$ is

$$Var(W) = (EW)^2\, c_W^2 \tag{3.56}$$

and

$$E\,(W^2) = Var\,(W) + (\,EW\,)^2 \tag{3.57}$$

Finally QNA calculates an approximate probability distribution for $W$. It is an atom at zero as seen from Equation (3.43) and has a density above zero. At this point, Whitt simply indicates that the density is chosen so that $W$ will have the first two moments already determined for them. In this case where $c_s^2 = 1$, Whitt defines $D$ with an exponential density with mean $ED$ [Whi83b].

QNA calculates the second moment and variance of $N$. The second moment is

$$E\,(N^2) = Var\,(N) + (EN)^2 \tag{3.58}$$

The Variance of $N$ is

$$Var\,(N) = (EN)^2\, c_N^2 \tag{3.59}$$

where $c_N^2$ is basically a correction factor to account for dividing by zero and for heavy traffic when $\rho \to 1$ causing the value $c_N^2 \to 1$. In this study, it reduces to the familiar

$$Var(N) = \frac{\rho}{(1-\rho)^2} \tag{3.60}$$

***3.6.4.2 GI/G/m Queue.*** QNA congestion formulations for the multi-server are exact for this study. The expected number of servers are calculated as

$$E\, min\{N,m\} = \alpha = \lambda\tau \tag{3.61}$$

which is exact for the M/M/m queue. The traffic intensity or utilization is also exact at

$$\rho = \frac{\alpha}{m} \tag{3.62}$$

Therefore, the congestion by Little's Formula:

$$EN = \alpha + \lambda\, EW \tag{3.63}$$

QNA only provides approximate *EW* formulations for non-M/M/m queues. The basic QNA modification is based on the standard *EW* for M/M/m queues given as *EW(M/M/m)*

$$EW\,(\, c_a^{\,2}\,,\, c_s^{\,2},\, m\,) = \left[\frac{(c_a^2 + c_s^2)}{2}\right] EW\,(M/M/m) \tag{3.64}$$

Whitt has been able to successfully show Equation (3.64) works well for M/G/m queues under heavy conditions and is asymptotically correct for GI/G/m systems as $\rho \rightarrow 1$ for a fixed m [Whi83b].

***3.6.5 Total Network Performance Measures.*** This study uses two levels of congestion measures, congestion at the system level and congestion seen by the primary customer or as Whitt describes it, a particular customer.

Within system level measures, QNA defines throughput as the total external arrival rate:

$$\lambda_0 = \lambda_{01} + \lambda_{02} + \ldots + \lambda_{0n} \tag{3.65}$$

Since no customers are created or combined at the nodes, the total external arrival rate equals the total departure rate from the network. If there was customer creation or combination, the total rate of departure from the network would be the same as Equation (3.20). In general, the overall congestion in the entire network is based on the mean and variance of the number of customers $N$.

$$EN = EN_1 + EN_2 + \ldots + EN_n \tag{3.66}$$

and assuming the nodes are independent, a Markovian model with a product-form solution approximates the variance to be

$$Var(N) = Var(N_1) + Var(N_2) + \ldots + Var(N_n) \tag{3.67}$$

The way QNA calculates the primary customer view is first to solve the equilibrium or overall behavior of the network. QNA then considers a particular customer dependent on which input method is used. In this study, the alternate input is used, QNA first converts the input into the standard input method. Next, QNA solves for the equilibrium behavior. Congestion measures are calculated for the class along the specified route. This is done with the assumption that upon arrival to each node, they see independent versions of the equilibrium state of the network.

Using the same notation as the alternate input method, the expected total service time for a customer of class $k$ (the primary customer) is

$$\sum_{j=1}^{n_k} \tau_{kj} \qquad (3.68)$$

which is the summation of all the service times of the n nodes on the $k$ th customer's route. This study considers the primary customer's route since it is the only customer with a route. The total expected waiting time is

$$\sum_{j=1}^{n_k} E(W_{n_{kj}}) \qquad (3.69)$$

which is the summation of the expected waiting times at the $n$ nodes of the $k$ th customer. The total travel time is the sum of Equations (3.68) and (3.69). In a similar manner, the variance of total service time is

$$\sum_{j=1}^{n_k} \tau_{kj}^2 c_{skj}^2 \qquad (3.70)$$

which is the summation of all the $n$ nodes' product form of the mean service time and the service time variability parameter of the $k$ th customer. The variance of the total waiting time is

$$\sum_{j=1}^{n_k} Var(W_{n_{kj}}) \tag{3.71}$$

which is the summation of all the waiting time variances of the $n$ nodes on the $k$ th customer's route. The variance of the total travel time is the sum of Equations (3.70) and (3.71).

Once QNA finishes analyzing the network, GNA allows the user to examine various performance parameters regarding the network. The bottleneck performance parameter shows nodes with traffic intensity at or greater than 0.8. The most significant parameter in this study is the route chosen by GNA that delivers the primary customer's traffic from the start node to the end node. The sequence of nodes forming this route along with the Total Sojourn Time are written to a text file for analysis by the user.

### 3.7 *Path Likelihood*

The choice for the GNA packet switching model, which produces a set of links for the primary customer's traffic, is based on relevant packet switching theory principles restricted by the limits of a simulation. In other words, this is only one sample based on a single execution of the GNA tool. The user repeats the GNA packet switching model a number of trials on the same network topology and primary customer traffic rate while varying BSTR and PTR values. Therefore, this tool can provide a percentage value of the possibility that a given link is in use by the primary customer. Ultimately, this is only an estimate of the actual network's mean value of specific link usage. From these mean link usage values, a set of links will form a segment that has a high probability of usage. There may be two or more segments that exist to deliver the traffic to a particular node. The segments with the highest probability of usage forming a route from the start node to the end node, which is the most likely path used. Therefore, this tool can be used as a specific link and path likelihood prediction tool.

### *3.8 Summary*

This chapter discussed the methodology used to produce the Specific Link and Path Likelihood Prediction Tool. The reader was provided an elementary overview with assumptions before proceeding to an in-depth look. This chapter explained how GNA was used to represent certain packet switching theory principles such as network representation and route determination. Two sample networks were presented that verify and empirically validate the GNA routing algorithm. Finally, this chapter reviewed the theory of QNA, central to the queueing analysis of this study.

In Chapter 4, Tool Analysis, a test network will be analyzed twice by the GNA tool. The tool will first examine any changes in specific link usage as the BSTR and PTR values vary. Afterwards, the tool will examine any changes in specific link usage after certain node characteristics are altered while the BSTR and PTR values are varied.

THIS PAGE INTENTIONALLY LEFT BLANK.

# *IV. TOOL ANALYSIS*

## *4.1 Introduction*

The goal of this chapter is to discover, through the use of the Specific Network Link and Path Likelihood Prediction Tool, which path and set of links are most likely to be used to deliver the traffic from the start to destination node when certain factors are altered. This study focuses on the Network Layer of the OSI model, along with GNA and QNA.

Once the general network information is entered by the user, QNA evaluates this network by analytically modeling the amount of congestion to be expected at each individual node. Given information of the network topology, source and destination nodes, GNA provides the most likely path for use based on the concepts used by Cheng and Lin [ChL95] in dealing with a path assignment and circuit routing problem in an ATM network. With this path data, GNA routes the traffic from the start node to the destination node. The user executes this tool once for each change in the factors. For this analysis, GNA routes the primary traffic through a test network, Network C. After each execution, the resulting path is appended to a text file for later analysis by the user. The topology of Network C is based on Figures 3.2a and 3.2b.

### *4.2 Parameters and Factors*

Each node of the 25 node test network has the two variables, the mean service time and service time variability parameter, as described in Chapter 3. This allows up to 50 independent variables in Network C. This analysis shows that a change in simply a few variables will alter the choice of link usage. Therefore, two versions of Network C, C1 and C2 are analyzed. Figure 4.1 combines both sets of links to show the complexity of the topology.

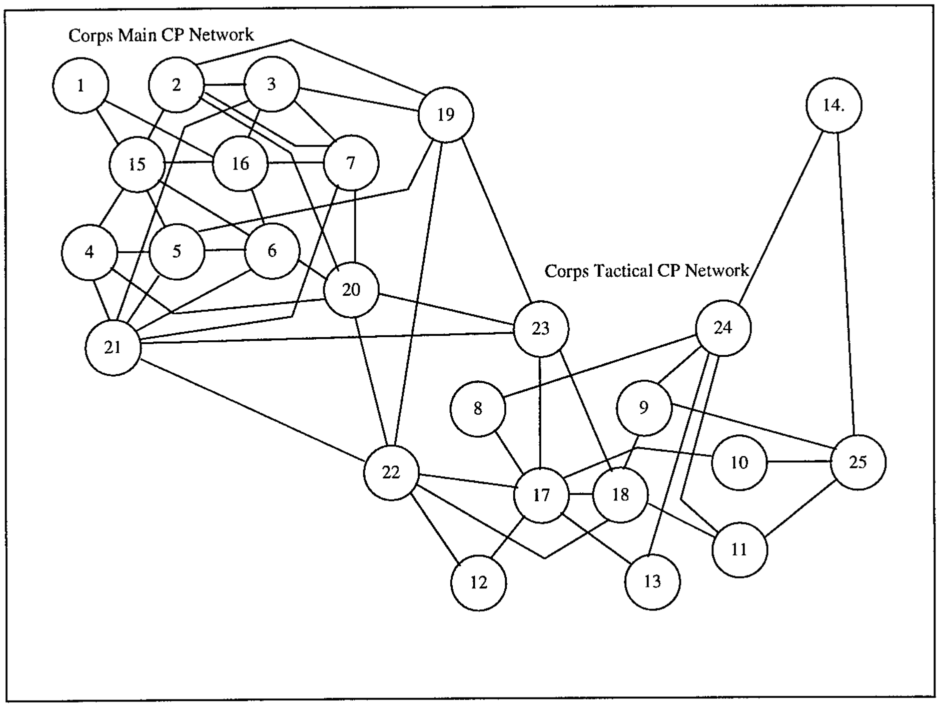


Figure 4.1 Combined External and Internal Links.

In each of the two versions, once a good quality of service (QoS) is achieved, the network flow control will admit the basic secondary traffic rate (BSTR) and the primary traffic rate (PTR) as the user defines them. A good QoS occurs when the user defined BSTR and PTR values do not cause the traffic intensity of any node to equal or exceed one. When any node's traffic intensity equals or exceeds one, GNA informs the user to reenter a lower BSTR or PTR value. In version C1, the parameters are the number of servers and the exact processing capability of the servers. In version C2, the server processing times for selected key nodes will change to test if the percentage of link usage changes. The selected nodes' service time adjustments occur in increments of 0.25. The increment changes stop at the values shown in Table 4.2 when route generation differs from the previous increment. In all cases, times are generic time units. Table 4.1 lists the node parameters, server capacities and server processing times for Network C1. Table 4.2 lists the parameters used in Network C2 with the changes underlined and in bold print.

Table 4.1 Parameters for Network C1.

| ID | Time | Servers | ID | Time | Servers | ID | Time | Servers | ID | Time | Servers | ID | Time | Servers |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.00 | 20 | 6 | 3.00 | 20 | 11 | 4.00 | 20 | 16 | 3.00 | 30 | 21 | 2.00 | 40 |
| 2 | 3.00 | 20 | 7 | 3.00 | 20 | 12 | 4.00 | 20 | 17 | 3.00 | 30 | 22 | 2.00 | 40 |
| 3 | 3.00 | 20 | 8 | 4.00 | 20 | 13 | 4.00 | 20 | 18 | 3.00 | 30 | 23 | 2.00 | 40 |
| 4 | 3.25 | 20 | 9 | 4.00 | 20 | 14 | 4.00 | 20 | 19 | 2.00 | 40 | 24 | 2.00 | 40 |
| 5 | 3.00 | 20 | 10 | 4.00 | 20 | 15 | 3.00 | 30 | 20 | 2.00 | 40 | 25 | 2.00 | 40 |

Table 4.2 Parameters for Network C2 with changes underlined.

| ID | Time | Servers | ID | Time | Servers | ID | Time | Servers | ID | Time | Servers | ID | Time | Servers |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.00 | 20 | 6 | 3.00 | 20 | 11 | 4.00 | 20 | 16 | 3.00 | 30 | # | **1.75** | 40 |
| 2 | 3.00 | 20 | 7 | 3.00 | 20 | 12 | 4.00 | 20 | 17 | 3.00 | 30 | # | **1.50** | 40 |
| 3 | 3.00 | 20 | 8 | 4.00 | 20 | 13 | 4.00 | 20 | 18 | 3.00 | 30 | 23 | 2.00 | 40 |
| 4 | 3.25 | 20 | 9 | 4.00 | 20 | 14 | 4.00 | 20 | 19 | 2.00 | 40 | 24 | 2.00 | 40 |
| 5 | 3.00 | 20 | 10 | 4.00 | 20 | 15 | 3.00 | 30 | 20 | 2.00 | 40 | # | **2.75** | 40 |

In both version C1 and C2, the factors are the primary and basic secondary traffic rates. These two factors are varied to represent a variety of realistic workloads.

GNA allows the highest BSTR based on a good QoS. The highest BSTR value varies according to the capabilities of the specific network. The highest allowable BSTR serves as the network's high load value. The low load factor is rated at 50% of the high load value. The medium load value is the mid-point value between the high and low values. Table 4.3 lists the specific BSTR values for Networks C1 and C2.

Table 4.3 BSTR Loading Level Values for Networks C1 and C2.

| Networks C1 and C2 | Low Load | Medium Load | High Load |
|---|---|---|---|
| BSTR values | **0.6** | **0.9** | **1.2** |

The PTR value acts as the second factor within the analysis. Network flow control will determine the highest PTR based on a good QoS. The PTR increments are 10% of the BSTR high value rounded to 2 significant digits. Given the primary customer's traffic is a higher priority than the basic secondary traffic, the PTR values are greater than the BSTR values [SaA94]. Therefore, the lowest PTR value is one increment above the BSTR value. A complete set of BSTR and PTR values form an experiment set. Each experiment set is composed of *n* executions (trials) of the GNA tool. The maximum value of *n* is the product of the cardinality of BSTR and the cardinality of PTR.

There are two forms for the cardinality of PTR. Therefore, there are two experiment sets. The first form starts the PTR values from one increment above the BSTR to twice the BSTR value. The second form starts the PTR value from one increment above the BSTR to the highest PTR value admitted by the network flow control. This second form takes into account the influence of higher values of the primary traffic. In the second form, the PTR increments are 20% of the BSTR high load rounded to two significant digits. This maintains an equal number of PTR values. Table 4.4a lists the PTR values for the first experiment set and Table 4.4b lists the

PTR values for the second experiment set. Both first and second experiment sets are used in Networks C1 and C2.

Table 4.4a PTR Values for the first experiment set.

| | | | | | |
|---|---|---|---|---|---|
| 1.3 | 1.4 | 1.5 | 1.6 | 1.7 | 1.8 |
| 1.9 | 2.0 | 2.1 | 2.2 | 2.3 | 2.4 |

Table 4.4b PTR Values for the second experiment set.

| | | | | | |
|---|---|---|---|---|---|
| 1.4 | 1.6 | 1.8 | 2.0 | 2.2 | 2.4 |
| 2.6 | 2.8 | 3.0 | 3.2 | 3.4 | 3.6 |

Tables 4.5a and 4.5b, showing both factors BSTR and PTR, determine the number of trials that are executed (4.5a for the first experiment set and 4.5b for the second experiment set). Each trial consists of one complete execution of GNA's packet switching option. Both yield a sequence of nodes describing the path used to deliver traffic from the start node to the end node. Tables 4.5a and 4.5b both show the values of the BSTR and PTR which fail to achieve a good

Table 4.5a BSTR and PTR Factors and Trials for the first experiment set.

| PTR\BSTR Values | 0.6 | 0.9 | 1.2 | 1.3 |
|---|---|---|---|---|
| 1.3 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.4 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.5 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.6 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.7 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 1.8 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 1.9 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.0 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.1 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.2 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.3 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.4 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.5 | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |

Legend: X = BSTR/PTR combination trial that achieved a good QoS. Bad-QoS = failure to achieve a good QoS, with the caused indicated by the last value: -BS = BSTR value fails the QoS. -P = PTR value fails the QoS.

QoS. Given the factors used, there are a total of 28 trials for the first experiment set and 22 trials for the second experiment set.

Table 4.5b BSTR and PTR Factors and Trials for the second experiment set.

| PTR\BSTR Values | 0.6 | 0.9 | 1.2 | 1.3 |
|---|---|---|---|---|
| 1.4 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.6 | **X** | **X** | **X** | **Bad-QoS-BS** |
| 1.8 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.0 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.2 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.4 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.6 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 2.8 | **X** | **X** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 3.0 | **X** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 3.2 | **X** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 3.4 | **X** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 3.6 | **X** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |
| 3.8 | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-P** | **Bad-QoS-BS** |

Legend: X = BSTR/PTR combination trial that achieved a good QoS. Bad-QoS = failure to achieve a good QoS, with the caused indicated by the last value: -BS = BSTR value fails the QoS. -P = PTR value fails the QoS.

### *4.3 Specific Link Analysis Techniques*

Each experiment set is analyzed by two techniques, the All Links-All Trials and the Stage Decomposition, to interpret the percentage of specific link usage. The Excel spreadsheet product calculates these values to three significant digits for all the data sets. Appendix C.1 for Network C1 and Appendix C.2 for Network C2 contain the outputs of these calculations.

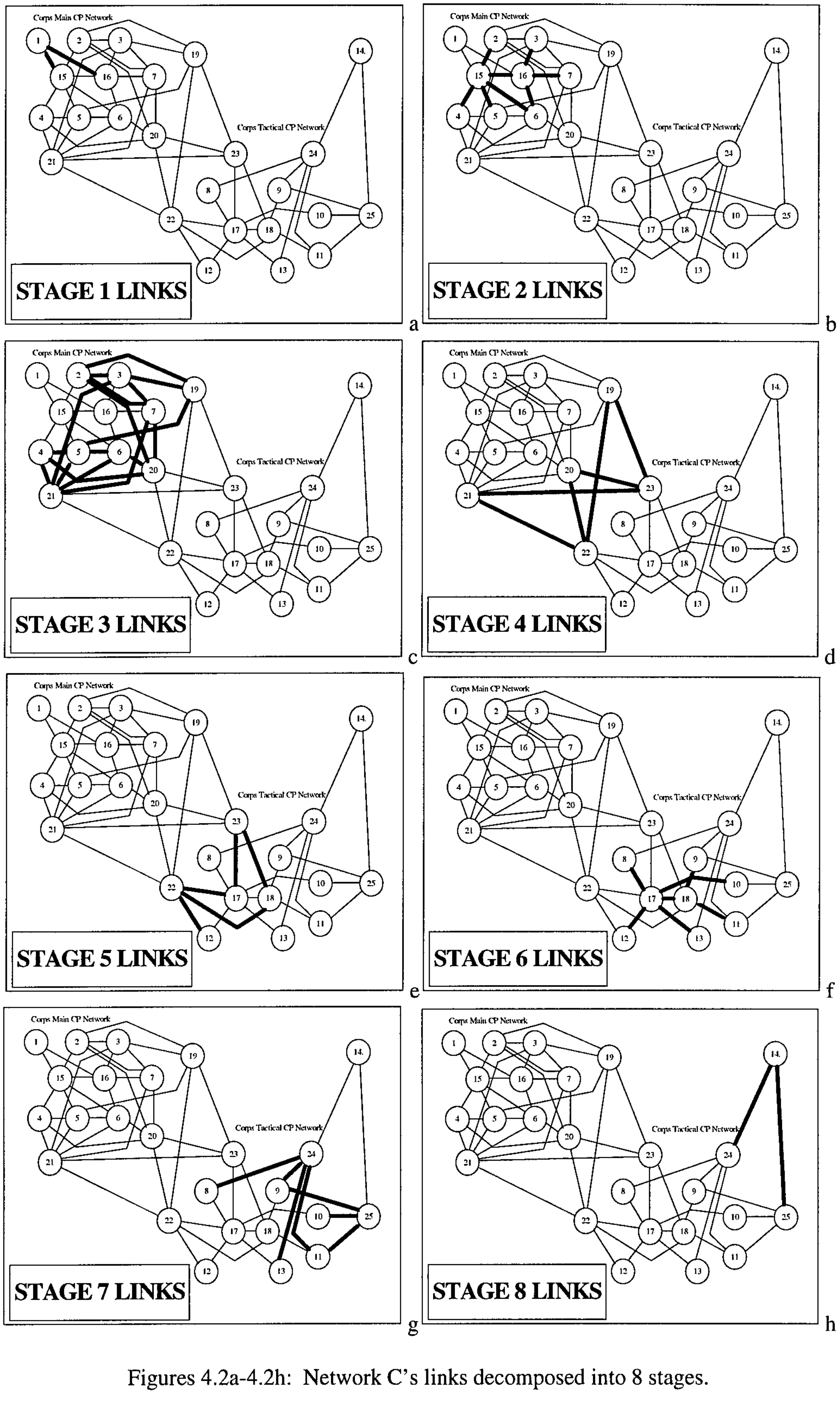
***4.3.1 First Technique.*** The first technique, All Links-All Trials, involves the percentage of a specific link usage compared to the total link usage among all the trials of the experiment set. Given Network C1 (refer to Figure 4.1) with node 1 as the start node and node 14 as the destination node, a possible node sequence of the first trial is 1-15-6-20-22-17-8-24-14. The second trial with different factors determines the node sequence as 1-16-3-21-23-18-11-25-14. The specific link 20-22 occurs once. The total link usage among both trials is 16 (Eight for the first trial and eight for the second trial). The specific link usage of 20-22 using the All Links-All Trials method is 6.25% (or 1/16).

***4.3.2 Second Technique.*** The second technique, Stage Decomposition, uses the tree traverse concept of depth [CoL90]. It requires a certain network topology where the source node is viewed as the root of the tree. The link from the source node leads to a number of child nodes. Each additional link increases the depth of the tree. This technique uses the term stages in place of depth levels to describe the links. Therefore, each link in the network can be systematically decomposed into stages starting from the source node.

Decomposition into stages involves placing the outgoing links from the start node into stage 1. The destination nodes at the end of the links in stage 1 are considered the candidate nodes for stage 2. Stage 2 contains the set of links that originate from the stage 2 candidate nodes and is actually used in a trial to deliver traffic, but are not already placed in stage 1. Therefore, the generic stage *n* contains the set of links that originate from the destination nodes in stage *n*-1 but are not already placed in any other stage prior to stage *n*. The links within stage *n* must also be used to deliver traffic in a trial. The last stage occurs when the destination node is the user-defined end node.

Once all trials are complete, an analysis of it determines whether the network topology can be decomposed. Not all network topologies can be completely decomposed into stages. For example, in the previous technique's two-trial execution, links 1-15 and 1-16 both are members of stage 1. Links 8-24 and 11-25 are members of stage 7. Both versions of Networks C are developed to completely decompose into eight stages. Figures 4.2a-4.2h depict all the possible links in Network C as members of one of the eight stages.

Within each stage, the percentage of specific link usage is determined by comparing the occurrence of a specific link to the total link membership. For example, in the previous



Figures 4.2a-4.2h: Network C’s links decomposed into 8 stages.

stage 1 example, the specific link 1-15 occurs once. The total link membership value in stage 1 is two (one for link 1-15 and one for link 1-16). The specific link usage of 1-15 using the stage decomposition technique is 50.0% (or 1/2). Usage of a specific link is appropriately compared among other links within the same stage. Given a network that can be completely decomposed into stages, this technique provides a more accurate depiction of specific link usage.

### *4.4 Network C1 Analysis*

Network C1 uses the parameters from Table 4.1 and the factors from Table 4.5a. Using Cheng and Lin's technique of grouping a set of physical links, where they refer to them as a macro logic link [ChL95], network C1 delivers the primary customer's traffic from node 1 to node 14 along four possible segments. Figure 4.3 depicts, in boldface the set of links used by the primary traffic. Each segment of the path is given an identifier for reference.

Analyzing the trials by the All Links-All Trials technique shows the usage ratio between Segments A and B at 6.0 to 1. The usage ratio between Segments C to Segment D is 1.3 to 1. Segments A and C, the links with a higher percentage of usage, are indicated by the solid lines in Figure 4.4. Segments B and D having a lower percentage of link usage are indicated by the dashed lines in Figure 4.4. With the same technique, the second experiment set also indicates that Segment A is used more often than Segment B by a ratio of 10.5 to 1, while Segment C is used over Segment D at a ratio of 3 to 2. Figure 4.5 graphically shows the path usage of the second experiment set. Table 4.6 summarizes the usage ratios between the segments of the two experiment sets using the All Links-All Trials technique.

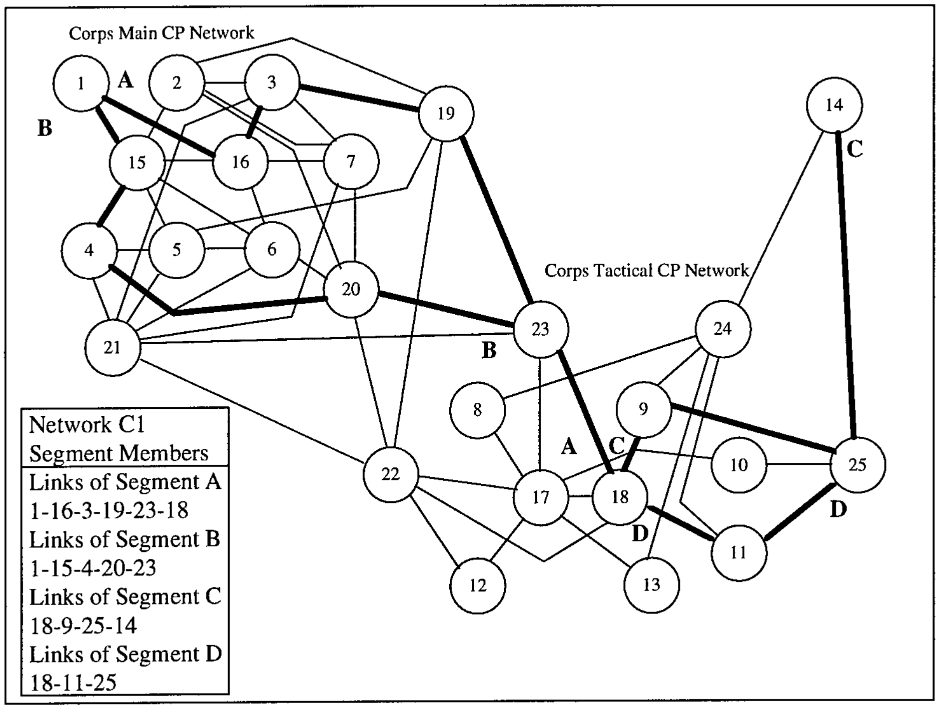


Figure 4.3 Primary Customer's 4 Segments in Network C1.

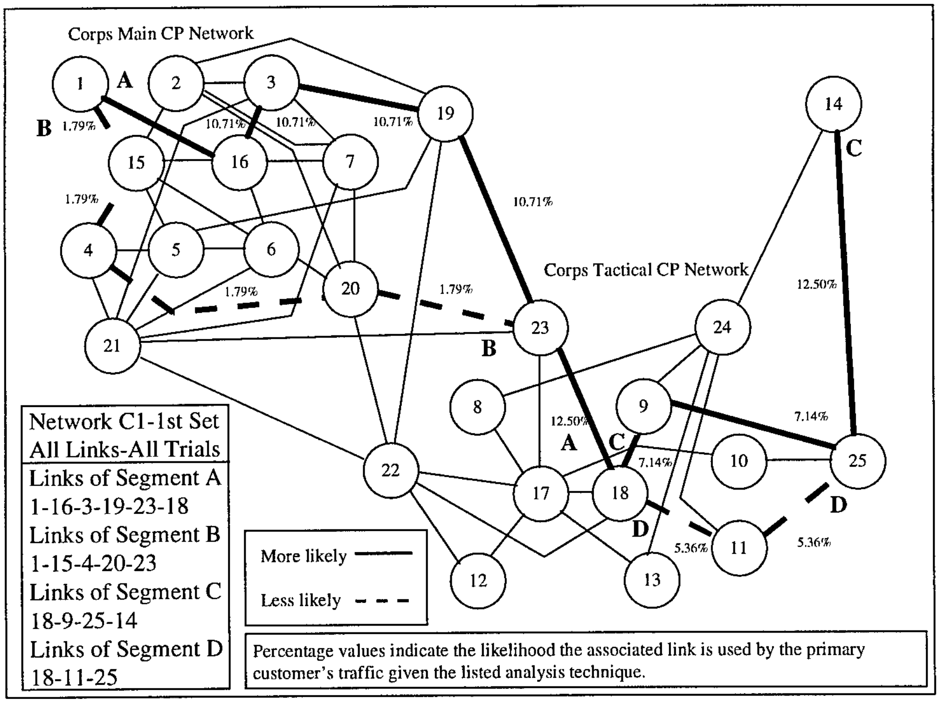


Figure 4.4 Network C1 first experiment set analyzed by the All Links-All Trials technique.

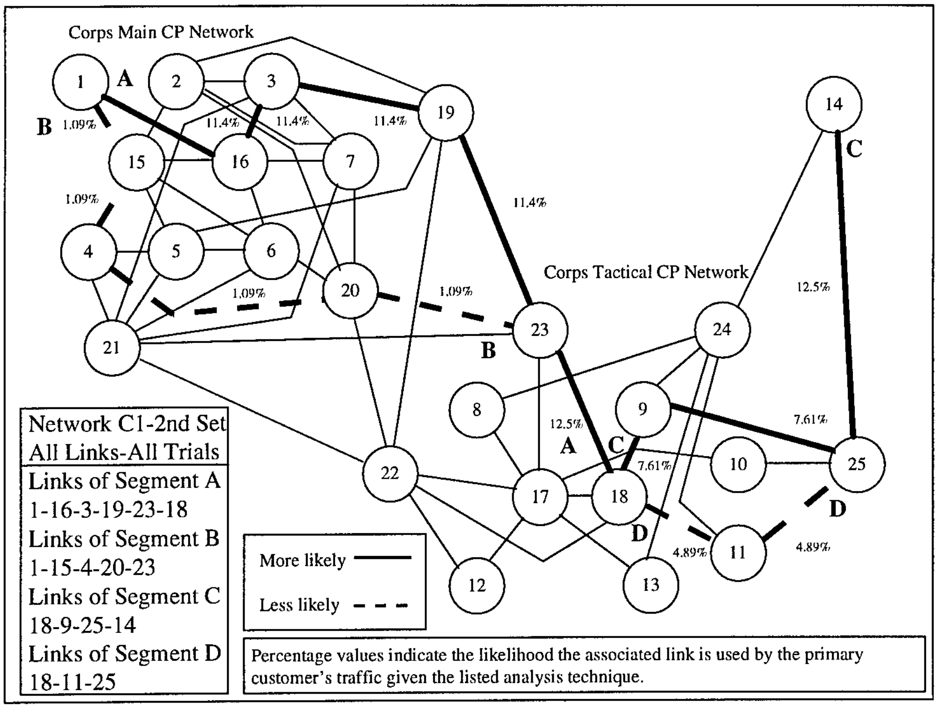
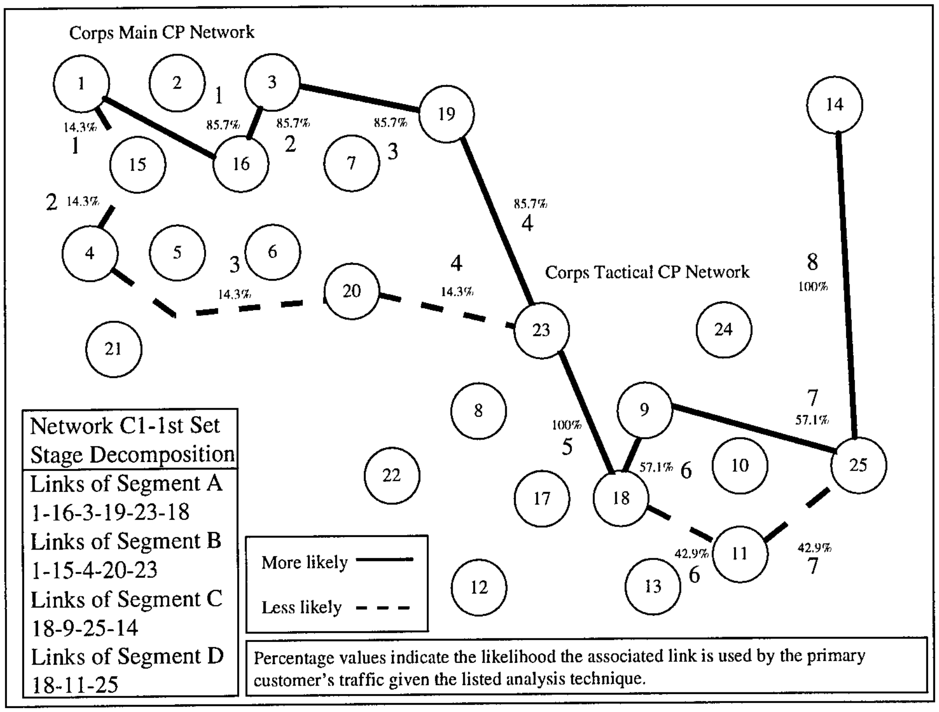


Figure 4.5 Network C1 second experiment set analyzed by the All Links-All Trials technique.

Table 4.6 Usage Ratio of Segments A to B and C to D using the All Links-All Trials technique.

| | 1st Experiment Set | | | 2nd Experiment Set | |
|---|---|---|---|---|---|
| Segment A | Segment B | Ratio | Segment A | Segment B | Ratio |
| 10.71% | 7.19% | 6.0 : 1 | 11.40% | 10.90% | 10.5 : 1 |
| Segment C | Segment D | Ratio | Segment C | Segment D | Ratio |
| 7.14% | 5.36% | 1.3 : 1 | 7.61% | 4.89% | 1.5 : 1 |

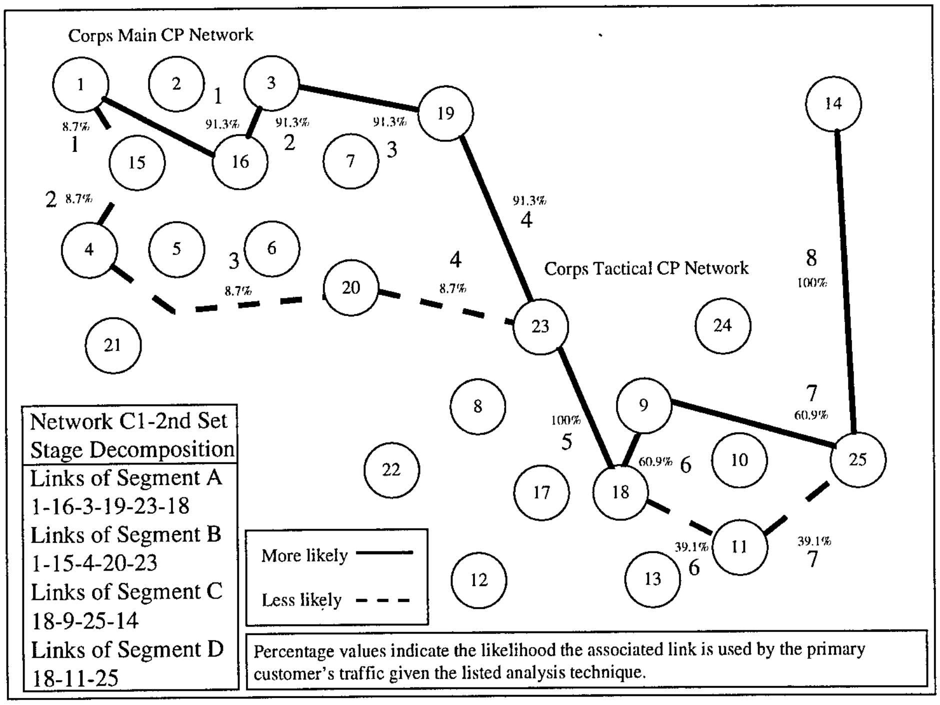
As indicated earlier, the second technique, Stage Decomposition, is applicable for this network topology and the given traffic conditions. In all the Stage Decomposition analysis figures, the unused links are removed for clarity. A stage group identifier also indicates the membership of each link. The first experiment set reveals that for every stage of the path, links with the higher usage are always a member of Segment A or C (Figure 4.6). The second experiment set also confirms the results of the first experiment set (Figure 4.7).



Network C1-1st Set Stage Decomposition
Links of Segment A
1-16-3-19-23-18
Links of Segment B
1-15-4-20-23
Links of Segment C
18-9-25-14
Links of Segment D
18-11-25

Percentage values indicate the likelihood the associated link is used by the primary customer's traffic given the listed analysis technique.

Figure 4.6 Network C1 first experiment set analyzed by the Stage Decomposition technique.



Network C1-2nd Set Stage Decomposition
Links of Segment A
1-16-3-19-23-18
Links of Segment B
1-15-4-20-23
Links of Segment C
18-9-25-14
Links of Segment D
18-11-25

Percentage values indicate the likelihood the associated link is used by the primary customer's traffic given the listed analysis technique.

Figure 4.7 Network C1 second experiment set analyzed by the Stage Decomposition technique.

Therefore, given the network conditions as specified with the parameters and the variety of workloads, it is approximately six to ten times more likely for the primary traffic to use the set of links in Segment A over B and approximately one and a half times more likely to use the set of links in Segments C over D. See Table 4.7 for a summary of the segment usage ratios. Although the specific link percentages are different, the usage ratios of both techniques are very similar.

Table 4.7 Usage Ratio of Segments A to B and C to D using the Stage Decomposition technique.

| | 1st Experiment Set | | | 2nd Experiment Set | |
|---|---|---|---|---|---|
| Segment A | Segment B | Ratio | Segment A | Segment B | Ratio |
| 85.70% | 14.30% | 6.0 : 1 | 91.30% | 8.70% | 10.5 : 1 |
| Segment C | Segment D | Ratio | Segment C | Segment D | Ratio |
| 57.10% | 42.90% | 1.3 : 1 | 60.90% | 39.10% | 1.5 : 1 |

### *4.5 Network C2 Analysis*

The analysis of Network C2 is accomplished using the same two techniques as in Network C1. Given the parameters from Table 4.2 and factors from Table 4.5b, Network C2 delivers the primary customer's traffic from node 1 to node 14 along three possible segments. Figure 4.8 depicts in boldface the set of links used by the primary traffic. As in Network C1, each segment of the path is given an identifier for reference. Note that in Network C2, there was a change in the processing capabilities of nodes 21, 22, and 25. The decrease in processing times at nodes 21 and 22 attempts to draw the primary traffic path through the selected nodes. Specifically, node 21 decreased by 12.5%, while node 22 decreased by 25%. The increase in processing time by 37.5% at node 25 attempts to push the primary traffic away from the selected node.

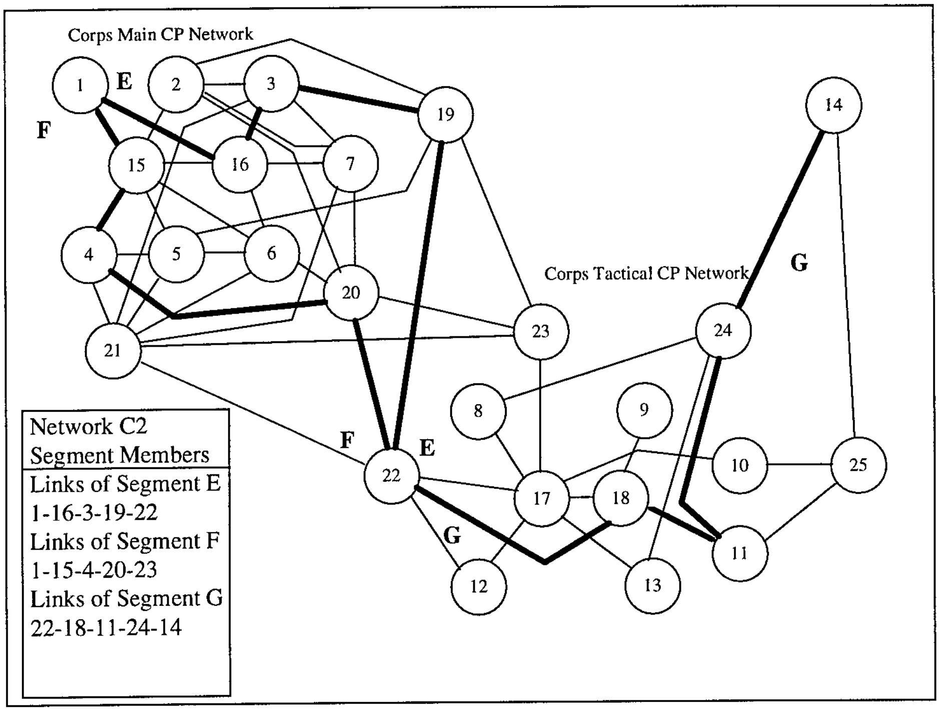


Figure 4.8 Primary Customer's 3 Segments in Network C2.

Analyzing the trials by the All Links-All Trials technique shows the usage ratio between Segments E and F at 6.0 to 1. There is no usage ratio beyond node 22. All travel occurs on the links of Segment G. Segment E, the links with a higher percentage of usage, is indicated by the solid lines in Figure 4.9. Segment F having a lower percentage of link usage is indicated by the dashed lines. With the same technique, the second experiment set, also indicates that Segment E is used more often than Segment F at a ratio of 10.5 to 1, while Segment G is used 100% of the time. Figure 4.10 graphically shows the path usage.

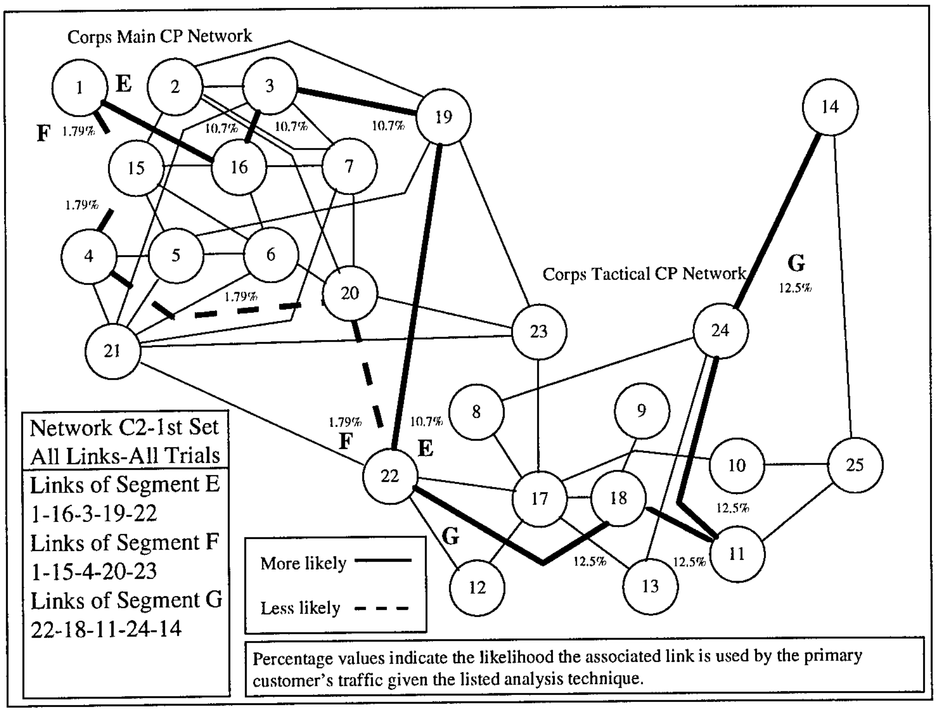


Figure 4.9 Network C2 first experiment set analyzed by the All Links-All Trials technique.

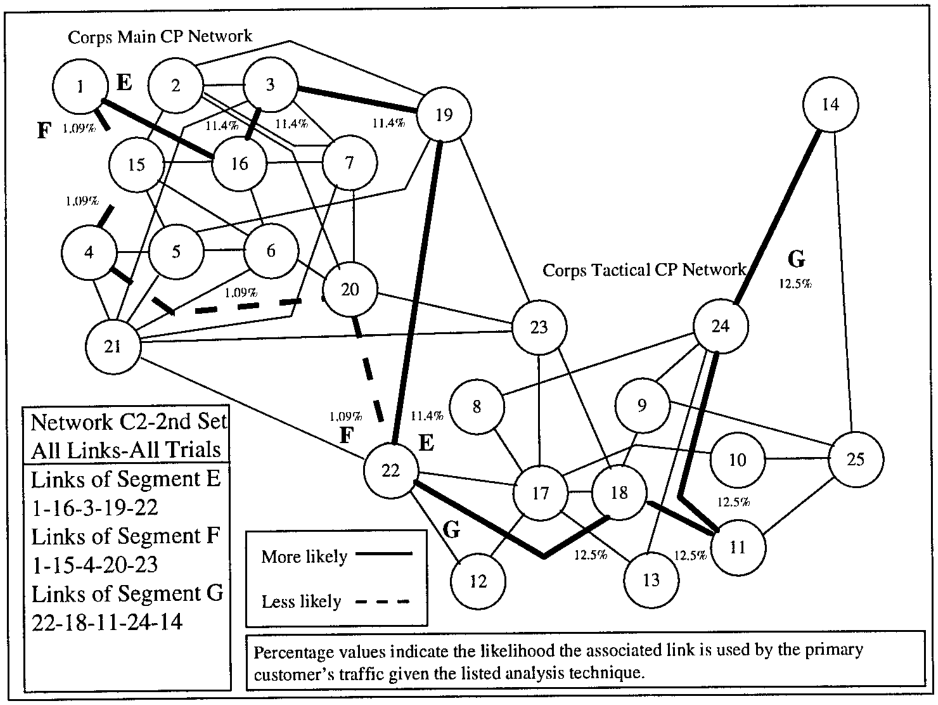


Figure 4.10 Network C2 second experiment set analyzed by the All Links-All Trials technique.

The altered nodes did affect the choice of the primary customer's path. The 25% decrease in processing time at node 22 draws all the primary traffic through it. Node 25, with the 32.5% increase in processing time, effectively diverts all the primary customer's traffic through the only other available node, node 24. Node 21 at first glance does not change the primary traffic in either the first or second experiment sets. Further analysis with the tool shows a change in the primary customer's traffic when the processing time decreased by another 12.5% of the original amount to a total of 25%. The text file output showing the change in the generated route is annotated in Appendix C.3. Table 4.8 summarizes the usage ratios between the segments of the two experiment sets using the All Links-All Trials technique.

Table 4.8 Usage Ratio of Segments E to F using the All Links-All Trials technique.

| | 1st Experiment Set | | | 2nd Experiment Set | |
|---|---|---|---|---|---|
| Segment E | Segment F | Ratio | Segment E | Segment F | Ratio |
| 10.70% | 1.79% | 6.0 : 1 | 11.40% | 1.90% | 10.5 : 1 |

Using the second analysis technique, the first experiment set reveals Segment E is still used over Segment F by a ratio of 6.0 to 1. Segment G's percentage does not change since the path is used exclusively (Figure 4.11). The second experiment set also supports the higher usage of Segment E over Segment F, but by a higher ratio of 10.5 to 1 (Figure 4.12). The effects of the altered nodes (22 and 25) are also reflected in this technique. The usage ratios using the Stage Decomposition technique produce very similar results as the All Links-All Trials technique (Table 4.9).

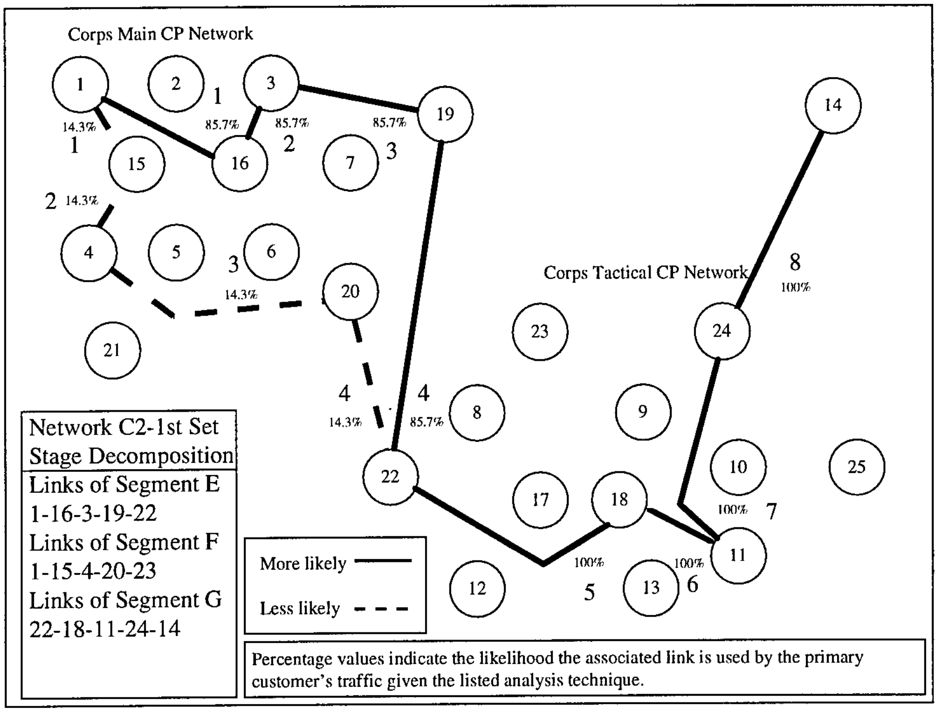


Figure 4.11 Network C2 first experiment set analyzed by the Stage Decomposition technique.

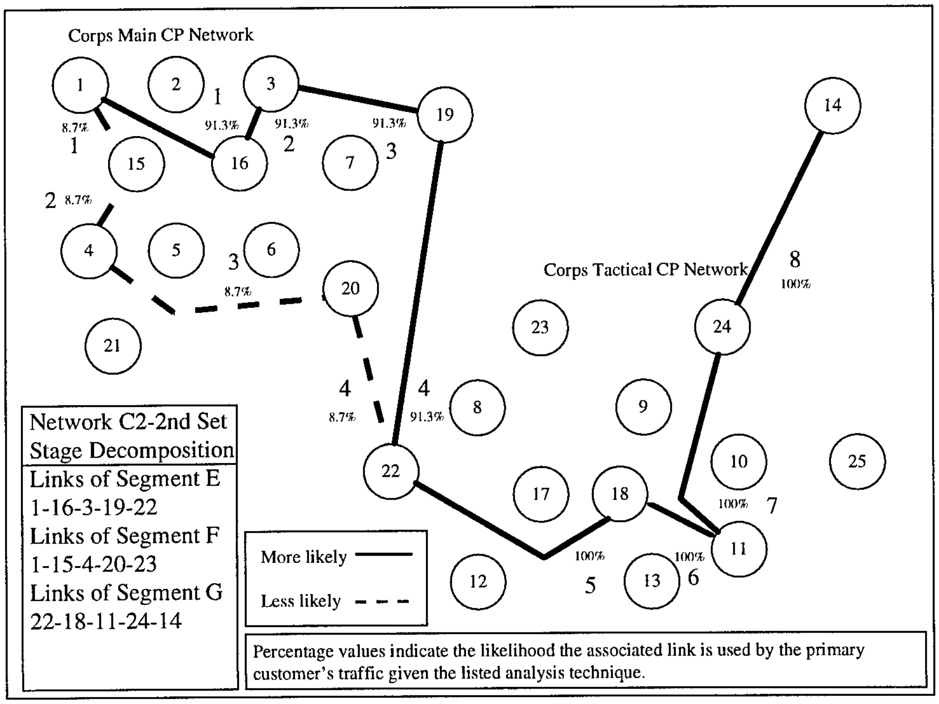


Figure 4.12 Network C2 second experiment set analyzed by the Stage Decomposition technique.

Table 4.9 Usage Ratio of Segments E to F using the Stage Decomposition technique.

| | 1st Experiment Set | | | 2nd Experiment Set | |
|---|---|---|---|---|---|
| Segment E | Segment F | Ratio | Segment E | Segment F | Ratio |
| 85.70% | 14.30% | 6.0 : 1 | 91.30% | 8.70% | 10.5 : 1 |

Therefore, given the changes to selected nodes between Network C1 and Network C2, the choice for the primary customer's path has changed to follow the faster processing node (22) and avoid the slower processing node (25). These changes also caused the choice of path beyond node 22 to completely rely on the links in Segment G. The changes for node 21 does not alter the path until the processing capability is increased by 25%.

### *4.6 Lower Bound of Sojourn Time*

The ideal lowest value in a customer's sojourn time occurs when the service, propagation, transmission, and waiting times within the servers at all nodes along the primary customer's path are zero. Given the current technology of communications, it is impossible. This section shows an analytical estimation of the lower bound on sojourn time given certain special conditions. Specifically, these conditions are no queueing at the servers, identical, independent, and exponentially distributed service rates, and identical mean service times of all servers involved.

With the sojourn time components used in this study, which are service and queueing times, the theoretical lowest values are zero in both instances. If the queueing time is zero, then the lower bound on sojourn time is just a function of the server's service time. QNA's sojourn time calculations take the product form, which takes into account the external arrival into each node and the departure streams from other nodes directed at the node [Whi83b].

However, in interpreting the meaning of the product form, independent nodal states do not imply the sojourn time at each node is independent [GeM80]. This interpretation also applies to servers of multiserver nodes [Die96]. Consider the two multiserver nodes in series with a single server node at the end in Figure 4.13.

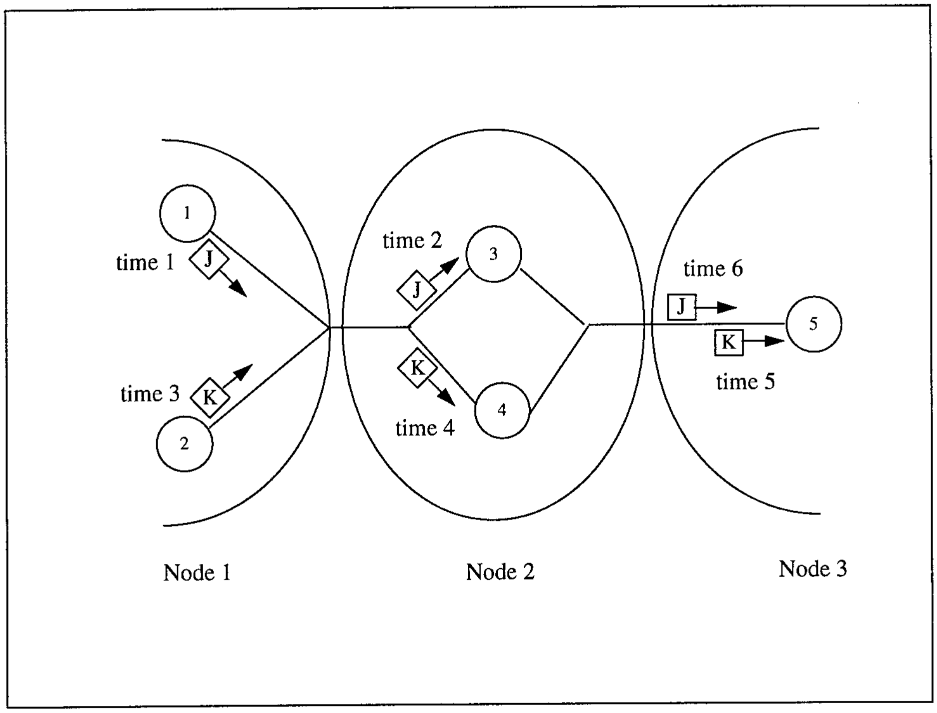


Figure 4.13 Independent servers in a node does not imply that the times are also independent.

Each server's service time is exponentially distributed. Suppose an attempt to calculate the lower bound of sojourn time is performed by using a value different than the mean service time. With a certain probability, server 1 in this instance is much faster at processing than another server 2 ($\mu_1 >> \mu_2$) and that server 4 is faster than server 3 ( $\mu_4 >> \mu_3$ ). Given another probability, let $J$ be a customer with a sojourn time at server 1; when $J$ leaves server 1, let be another customer $K$ with a longer sojourn time leave server 2. As $J$ enters server 2, and $K$ enters server 3, $K$ will complete service at 3 and proceed to server 4 at the next node. Customer $J$, which completes service at server 1 with a relatively shorter sojourn time than $K$, will complete service at server 4 with a relatively longer sojourn time than customer $K$. Therefore, the conditional probability of a longer total sojourn time at server 5, given a longer sojourn time at server 1, is not independent. While estimating the lower bounds, this section uses the mean service times of the nodes along the primary customer's path as calculated by GNA. Therefore, this problem is not an obstacle [Die96].

Whitt indicates the QNA assumption that each node within the network is seen as independent and identically distributed [Whi83b]. The parameters GNA are given describe all the servers as exponentially distributed. QNA calculates the customer's total sojourn time through all the nodes by summing all their service times. Given this, the Central Limit Theorem applies as each node is viewed as $X_1, X_2, X_3, \ldots, X_n$ and are independent random variables which are identically distributed. They each have a finite mean service time $\tau$ and a variance $\sigma$. Then if $S_n = X_1 + X_2 + \ldots + X_n$ ( n = 1, 2, ...),

$$\lim_{n\to\infty} P[a \leq \left(\frac{S_n - n\tau}{\sigma\sqrt{n}}\right) \leq b] = \frac{1}{\sqrt{2\Pi}} \int_a^b e^{\frac{-u^2}{2}} du \qquad (4.1)$$

where $\left(\frac{S_n - n\tau}{\sigma\sqrt{n}}\right)$ the standardized variable corresponding to $S_n$, is asymptotically normal [Spi75].

Since $S_n$ is behaving asymptotically normal, the QNA calculation for total sojourn time will approach the $S_n$ value. If the total sojourn time can asymptotically approach the normal, the Normal or Gaussian density function can completely describe the distribution for any value of $S_n$.

Rather than discovering the probability $P$ of the variable $S_n$, this section estimates at a constant probability $P$ that the Total Sojourn Time is less than the lower bound, the variable $S_n$. This also means that at probability *(1-P)*, the Total Sojourn Time is greater than the lower bound, $S_n$. Since the standard normal distribution can be completely described, the approximate inverse transformation of the standard normal is

$$\text{Standardized Variable} \cong \frac{(P^{0.135}) - (1-P)^{0.135}}{0.1975} \tag{4.2}$$

Equation (4.2) has a one decimal accuracy when the values of $0.0013499 \leq P \leq 0.9986501$ [Klo96]. Note when the value of P = 0.5, the value of $S_n$ is simply $n^*\tau$ or the standard QNA calculation for the primary customer's total sojourn time. Substituting the standardized variable into Equation (4.2), the lower bound estimation of $S_n$ occurring with probability $P$ is

$$\min(S_n) \text{ at probability } P = \sigma\sqrt{n}\left(\frac{P^{0.135} - (1-P)^{0.135}}{0.1975}\right) + n\tau \tag{4.3}$$

This estimation uses the standard normal distribution, it is mathematically possible to calculate a $S_n$ value equal to less than zero. All negative values should be treat as zero.

***4.6.1 Lower Bound Including Delay.*** A more special case situation can yield a tighter lower bound estimation. Given all the nodes involved in the primary route are M/M/1 queues, where the service time is an exponential distribution and the waiting time is a truncated exponential distribution [Jai91], this case assumes the sojourn times for each node has a mean, $\tau_k$ and standard deviation $\sigma_k$ . Given this assumption for $S_n$ (the sum of the individual sojourn times) to be asymptotically normal, the general conditions under the Central Limit Theorem will also apply. Therefore in Equation (4.3), $\sum_{k=1}^{n}\tau_k$ can be substituted for $n\tau$ and $\left(\sum_{k=1}^{n}\sigma_k^2\right)^{1/2}$ is substituted for $\sigma\sqrt{n}$ [All90]:

$$\min(S_n) \text{ with delay at probability } P = \left(\sum_{k=1}^{n}\sigma_k^2\right)^{1/2}\left(\frac{P^{0.135}-(1-P)^{0.135}}{0.1975}\right)+\sum_{k=1}^{n}\tau_k \qquad (4.4)$$

While the lower bound estimation provides the probability of a fast sojourn time, upper bound information provides a probability of a maximum for total sojourn time.

### *4.7 Upper Bound of Sojourn Time*

This section provides two estimation techniques of the upper bound of sojourn time. The first technique provides parameters for an expected upper bound on sojourn time. These parameters can be used in the GNA tool. The second technique is a special case when the nodes along the primary customer's path are M/M/1 queues. Since the upper bound is dependent upon the path chosen through the network, a complete solution does not exist until after the routing algorithm has determined a path.

***4.7.1 First Technique.*** The first upper bound estimation technique is a function of both the BSTR and the PTR values. Therefore, the upper bound will be based on the maximum BSTR and PTR allowed by the network flow control. The flow control accepts the BSTR value entered by the user when a good QoS occurs. A good QoS occurs when the traffic intensity of each node in the network is less than one. QoS fails when the traffic intensity of any node in the network is equal to or greater than one. The flow control does not accept the BSTR values leading to this condition.

An M/M/m node's traffic intensity is a function of the external arrival rate and the node's processing capability [Jai91]

$$\rho = \frac{\lambda}{m\mu} \tag{4.5}$$

where the value of $\lambda$ is determined by the number of links $\ell$ times the BSTR

$$\lambda = \ell(BSTR)\,. \tag{4.6}$$

Combining Equations (4.5) and (4.6), the value of BSTR is

$$BSTR = \frac{m\mu}{\ell}\rho \tag{4.7}$$

where $0.000001 \le \rho \le 0.999999$. It is possible to determine the maximum allowable BSTR for the entire network by calculating Equation (4.7) for each node $i$ in the network, where the traffic intensity $\rho$ approaches the value of one

$$\max(BSTR_i) = \frac{m_i \mu_i}{\ell_i} \rho \qquad (4.8)$$

where ρ = 0.999999. The number of significant digits in ρ is a function of the QNA output format. The max(BSTR) accepted for the network is the lowest value of the max(BSTR$_i$) of all nodes $i$. This maximum value of BSTR is valid regardless of whether or not the node limiting it is chosen as part of the primary customer's path. It is often the case, the node in question will not be chosen by the routing algorithm as a part of the primary customer's path. Since the node's traffic intensity value ρ is 0.999999, the node's sojourn time is increased and therefore avoided by the routing algorithm.

The maximum PTR value allowed depends on the path chosen to deliver the primary customer's traffic. The routing algorithm must determine a path before the remaining upper bound estimation solution can be found. The maximum PTR value is determined in a similar method as the maximum BSTR. The flow control accepts the PTR value entered by the user when a good QoS occurs. In a similar manner, the QoS fails when the traffic intensity of any node in the path of the primary customer is equal to or greater than one. As with the BSTR value, flow control does not accept the PTR value leading to this condition.

The traffic intensity formula in each node on the path is nearly the same as Equation (4.5). The change from Equation (4.5) occurs with the value of λ. The calculation of λ for node $j$ is the same as the QNA value of λ Equation (3.3), where $\lambda_{0j}$ is the arrival rate of the secondary traffic into the node. The value of $\sum_{i=1}^{n} \lambda q_{ij}$ is equivalent to the PTR value, and when $q_{ij}$ is one when $i$ is the previous node to $j$ on the primary customer's path. Therefore

$$\lambda_j = \ell_j \max(BSTR) + PTR \tag{4.9}$$

where $\ell$ is the number of links associated with node $j$ and max($\text{BSTR}_i$) is the maximum value of BSTR accepted. Combining Equations (4.5) and (4.9), the maximum value of PTR for each node $j$ is

$$\max(PTR_j) = m_j \mu_j \rho - \ell_j \max(BSTR_i) \tag{4.10}$$

where $\rho = 0.999999$. The max(PTR) accepted for the network is the lowest value of the max($\text{PTR}_j$) of all nodes $j$.

Using the max(BSTR) and the max(PTR), the traffic intensity for each node $j$ in the primary customer's path can be determined as

$$\rho_j = \frac{\ell_j \max(BSTR) + \max(PTR)}{m_j \mu_j} \tag{4.11}$$

The expected upper bound on sojourn time, max *E(ST)*, along the primary customer's path of $n$ nodes can be determined as

$$\max E(ST) = \sum_{j=1}^{n} \frac{1}{\mu_j}\left(1 + \frac{(1-Q_j)}{m_j(1-\rho_j)}\right) \tag{4.12}$$

The symbol $Q_j$ is the probability of queueing at each node $j$ with an M/M/m queue [Jai91] is

$$Q = \frac{(m\rho)^m}{m!(1-\rho)} P_0 \qquad (4.13)$$

where $P_0$ is the probability of zero customers in the system which is

$$\mathrm{P}_0 = [1 + \frac{(m\rho)^m}{m!(1-\rho)} + \sum_{n=1}^{m-1} \frac{(m\rho)^n}{n!}]^{-1} \qquad (4.14)$$

Therefore, the max(BSTR) and max(PTR) values are the parameters leading to the expected upper bound of final sojourn time. The user can apply these values with the GNA tool to provide an expected upper bound on sojourn time.

***4.7.2 Second Technique.*** The second upper bound estimation technique provides a sojourn time at a $q$-percentile probability for all nodes' service and waiting times. It requires the special condition that the nodes along the primary customer's path are M/M/1 queues. This can be seen as a single network of single processor computer workstations. This effectively represents a small LAN using the standard traffic and service generation models (an exponential distribution).

The special requirement allows the node's cumulative distribution function of service time $\tau$ to be exponentially distributed,

$$F(\tau) = 1 - e^{-\tau\mu(1-\rho)} \qquad (4.15)$$

while it is not the case in an M/M/m queue [Jai91]. The cumulative distribution function of waiting time, (W), is a truncated exponential distribution [Jai91]

$$F(W) = 1 - \rho e^{-W\mu(1-\rho)} \tag{4.16}$$

Since the exponential distribution can be described, the $q$-percentile of service time, $r_q$, can be calculated as follows:

$$1 - e^{-r_q\mu(1-\rho)} = \frac{q}{100} \tag{4.17}$$

or

$$r_q = \frac{1}{\mu(1-\rho)} \ln\left(\frac{100}{100-q}\right) \tag{4.18}$$

The $q$-percentile of waiting time, $W_q$, can be calculated as follows:

$$W_q = \frac{1}{\mu(1-\rho)} \ln\left(\frac{100\rho}{100-q}\right) \tag{4.19}$$

Combining Equations (4.18) and (4.19) results in the sojourn time at the $q$-percentile for both service and waiting times at node $j$

$$ST_j(q) = \frac{1}{\mu_j(1-\rho_j)} \ln\left(\frac{100}{100-q}\right) + \frac{1}{\mu_j(1-\rho_j)} \ln\left(\frac{100\rho_j}{100-q}\right) \tag{4.20}$$

Using the same assumptions Whitt used for developing QNA, that each node is treated independently, the final sojourn time along the primary customer's path can be found as:

$$ST(q) = \sum_{j=1}^{n} \left( \frac{1}{\mu_j(1-\rho_j)} \ln\left( \frac{100}{100-q} \right) + \frac{1}{\mu_j(1-\rho_j)} \ln\left( \frac{100\rho_j}{100-q} \right) \right) \tag{4.21}$$

Lacking this assumption, the only known case requires a specific network topology. The specific topology occurs when *n* nodes are strictly in tandem. All except the first and last nodes contain a single exponential server. The first node is an M/M/k queue and the last can be an M/G/k queue [GeM80].

### *4.8 Network D*

Network D is a test network developed to show the application of the two analytical techniques developed from this chapter. The GNA tool produced the sojourn times of Network D. Network D comprises 16 single-processor workstations in a four by four multi-stage switching topology. Figure 4.14 shows Network D arranges as a Banyan-type network with connections layered in an extra stage cube fashion [Rai96b].

Each node has a single server with a common exponentially distributed, mean service time of two. Since modeling traffic accurately is continually under research [Rai96a], this study uses three representative traffic levels. This study accomplishes this by using a combination of BSTR

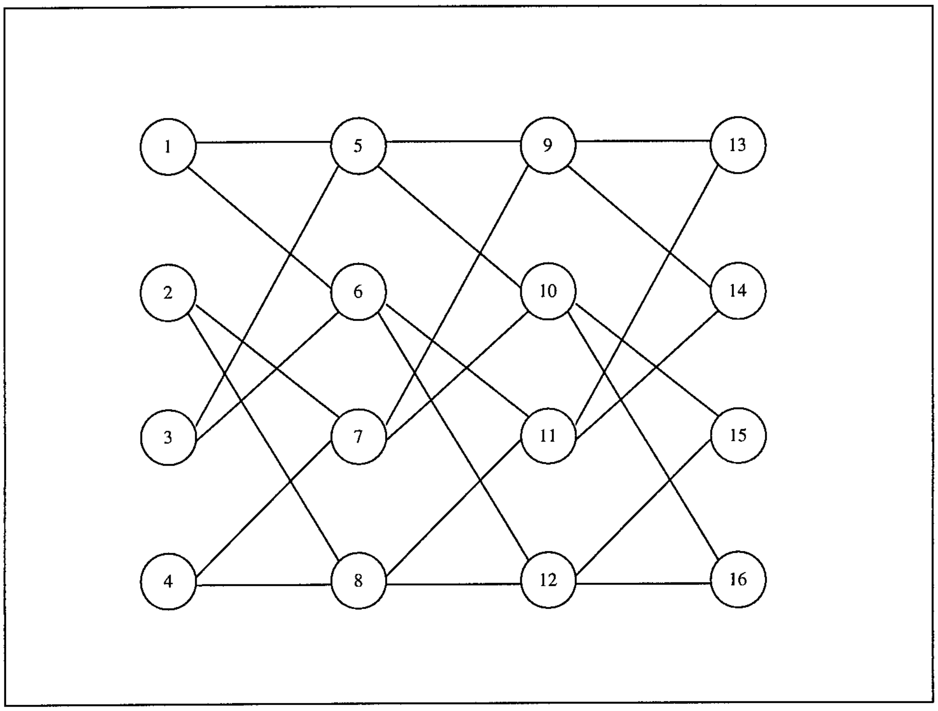


Figure 4.14 Network D arranged as a Banyan-type network with connections layered in an extra stage cube

and PTR values (Table 4.10). A Low intensity trial aims toward a combination of low PTR and BSTR values. A High intensity trial aims toward a combination of high PTR and BSTR values without causing traffic intensities to equal one or greater. A medium intensity trial is a compromise between the values of the Low and High intensity trials. These three intensity levels provide a good spread of traffic intensity values, ρ, ranging from 0.040 to 0.960 (Table 4.11).

Table 4.10 Representative PTR and BSTR Values for Network D

| Trial Intensity | PTR Values | BSTR Values |
|---|---|---|
| Low | 0.01 | 0.03 |
| Medium | 0.10 | 0.06 |
| High | 0.18 | 0.09 |

Table 4.11 Traffic Intensities and Sojourn Times of Network D

| Trial Intensity | ρ at first node | ρ at second node | ρ at third node | ρ at fourth node | Sojourn Time |
|---|---|---|---|---|---|
| Low | 0.060 | 0.080 | 0.096 | 0.040 | 8.6376 |
| Medium | 0.440 | 0.480 | 0.621 | 0.240 | 15.9810 |
| High | 0.720 | 0.720 | 0.960 | 0.360 | 77.0190 |

Using the four nodes' mean service time value, $\tau = 2$, and a variance value, $\sigma = 4$, the Total Sojourn Time being greater than the lower bound can be determined at a probability of *(1-P)*. This value of *(1-P)* indicates the probability that the Total Sojourn Time is greater than the lower bound. Using the lower bound with delay technique previously described, Equation (4.4) from Section 4.6.1 and a *P* value of 0.20, the lower bound with delay value is calculated as 5.0330 time units for the low traffic representative trial. This means users have a 80% (or 1 - *P*, where *P* = 0.20) assurance the Total Sojourn Time of the Low intensity trial is greater than this lower bound value.

Appendix D.1 contains the output of the Excel spreadsheet program showing various calculated lower bound values for all three intensity levels. Table 4.12 contains a summary of these calculations. It shows the level of certainty, for values calculated from Equation (4.4), can properly act as the lower bound value for the Total Sojourn Time.

Table 4.12 Various Lower Bounds with Delay Probabilities for All Three Trials

| Probability of Sojourn Times being greater | Lower Bound (Low) | Lower Bound (Medium) | Lower Bound (High) |
|---|---|---|---|
| 80% | 5.0330 | 9.0422 | 26.3239 |
| 70% | 6.3957 | 11.6654 | 45.4892 |
| 60% | 7.5556 | 13.8980 | 61.8006 |

With a start node of 1 and an end node of 15, the Total Sojourn Times are listed in Table 4.11. Given the low traffic representative trial in Network D, with a PTR value of 0.01 and a

BSTR value 0.03, the values in Table 4.12 correctly bounds the mean sojourn time determined by the GNA tool.

Note the lower bound of the Low Intensity trial as it increases, the assurance to properly bound the Total Sojourn Times has decreased. This is seen in all three trials of the representative traffic levels. Equation (4.1) determines the probability of the standardized variable being greater than the Total Sojourn Time distributed in the Standard Normal fashion. Increasing the lower bound value corresponds to increasing the variable $S_n$ within Equation (4.1). Increasing the $S_n$ variable increases the standardized variable $\left(\frac{S_n - n\tau}{\sigma\sqrt{n}}\right)$. Given the inverse exponential contribution of the standardized variable towards the integration, increasing the standardized variable increases the total integral, but at a decreasing rate. Despite the decreasing rate, the total integral of Equation (4.1) will increase signifying the increase probability of the term $S_n$ being greater than the normally distributed Total Sojourn Time. Given the probability for all events must sum to one, increasing this probability consequently decreases the assurance of the lower bound value properly bounding the Total Sojourn Times.

Figure 4.15 gives an excellent depiction of the apparent linear relationship between the increase in the bounding probability and the lower bound value of the high traffic level. Given this model's assumptions, designers can readily see the point of diminishing return for the increased lower bound to properly bound the Total Sojourn Times when the traffic load is high. Of greater note is the relationship when the traffic level conditions are at medium or less. Given low to medium loads, a small increase of the lower bound greatly reduces the probability to properly bound the Total Sojourn Times. Designers should note when information travels under low to medium conditions, it encounters less congestion and attempts to tighten the lower bound by increasing it will not benefit the design specifications.

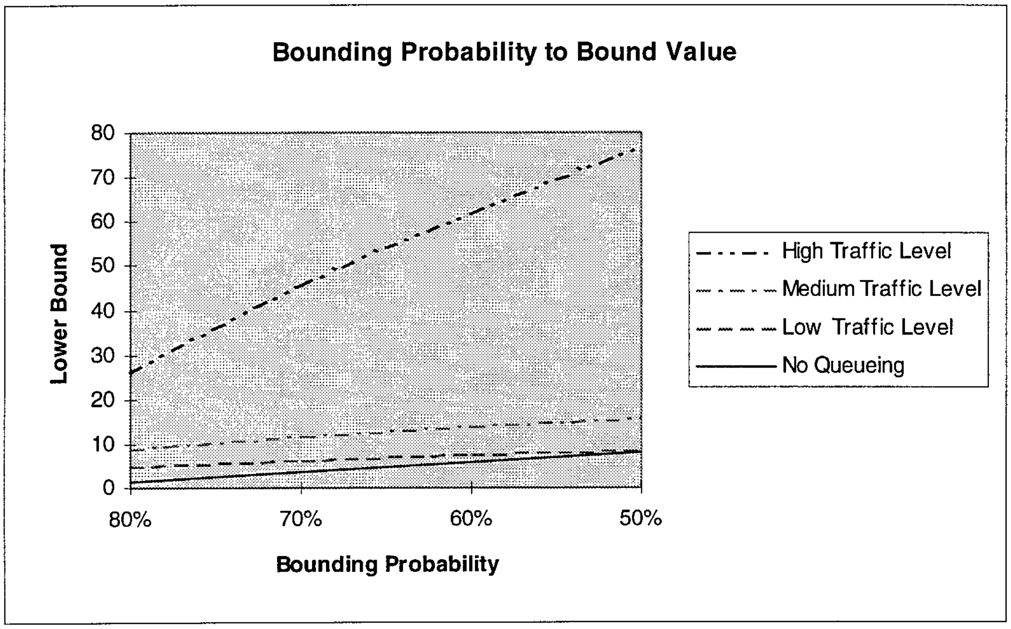


Figure 4.15 Lower Bounding Probability to Lower Bound Value

Inspecting Equation (4.4) explains the cause to the decreasing assurance more clearly. The term $\sum_{k=1}^{n} \tau_k$ refers to the summation of the mean service times of the nodes. As the nodes do not change, this implies the parameters of nodes are held constant. This term cannot contribute towards the lower bound value. Of the two terms involved in Equation (4.4), the remaining term $\left(\sum_{k=1}^{n} \sigma_k^2\right)^{1/2}\left(\frac{P^{0.135}-(1-P)^{0.135}}{0.1975}\right)$ must cause the decreased assurance. The factor $\left(\sum_{k=1}^{n} \sigma_k^2\right)^{1/2}$ refers to the square root taken over the summation of the nodes' variance values. Similar to the reason applied to the $\sum_{k=1}^{n} \tau_k$ term, the factor $\left(\sum_{k=1}^{n} \sigma_k^2\right)^{1/2}$ cannot contribute to the decreased assurance. The remaining factor $\left(\frac{P^{0.135}-(1-P)^{0.135}}{0.1975}\right)$ contains the term $(1\text{-}P)^{0.135}$, which specifies the assurance of bounding. Given the inverse relationship between the term $(1\text{-}P)^{0.135}$ and term $(P)^{0.135}$ , increasing the term $(P)^{0.135}$ directly decreases the term $(1\text{-}P)^{0.135}$ . As the term $(P)^{0.135}$ is increased to raise the lower bound value, consequently, the corresponding term $(1\text{-}P)^{0.135}$ for assurance is decreased.

An attempt to increase value of the $S_n$ by increasing the other terms $n\tau$ and $\sigma\sqrt{n}$ instead of the term $\left(\frac{P^{0.135}-(1-P)^{0.135}}{0.1975}\right)$ can be argued. Unfortunately, increasing any values of $n$, $\tau$ , or $\sigma$ ultimately cause the mean Total Sojourn Time to increase. This increase of the mean signifies the bulk of normally distributed Total Sojourn Times on a *pdf* graph has migrated towards higher values. This migration counters the increasing lower bound value. The next section shows the change in lower bound as the number of nodes increases while maintaining the same probability of bounding.

Inspecting Equation (4.4), it can be seen as the number of nodes $n$ increases, it directly increases the lower bound value without a need to change the probability *P*. Appendix D.2 contains the output of the Excel spreadsheet program showing various calculated lower bound with delay values as the number of nodes, $n$ , increases for all three intensity levels. Table 4.13 contains a summary of these calculations with the probability equal to 80%. Therefore, this

Table 4.13 Lower Bound Values for All Three Trials at 80% Probability

| **Number of Nodes** | **Lower Bound (Low)** | **Lower Bound (Medium)** | **Lower Bound (High)** |
|---|---|---|---|
| 4 | 5.0330 | 9.04216 | 26.3240 |
| 8 | 13.6706 | 25.0232 | 103.3429 |
| 10 | 17.9894 | 33.0137 | 141.8524 |
| 20 | 39.5834 | 72.9662 | 334.3999 |
| 40 | 82.7714 | 152.8712 | 719.4949 |

estimation technique increases the lower bound as the number of nodes visited increases without decreasing the level of certainly.

Table 4.14 also depicts the number of nodes increasing for each level by two. With the exception when the number of nodes are 4, there is at least a rough doubling of the lower bound value per doubling of the number of nodes. Figure 4.16 shows this rough doubling graphically.

Table 4.14 Lower Bound Values When Doubling Nodes for All Three Trials at 80% Probability

| Number of Nodes | Lower Bound (Low) | Lower Bound (Medium) | Lower Bound (High) |
|---|---|---|---|
| 4 | 5.0330 | 9.0422 | 26.3240 |
| 8 | 13.6706 | 25.0232 | 103.3429 |
| 16 | 30.9458 | 56.9852 | 257.3810 |
| 32 | 65.4962 | 120.9092 | 565.4569 |
| 64 | 134.5970 | 248.7572 | 1181.6089 |

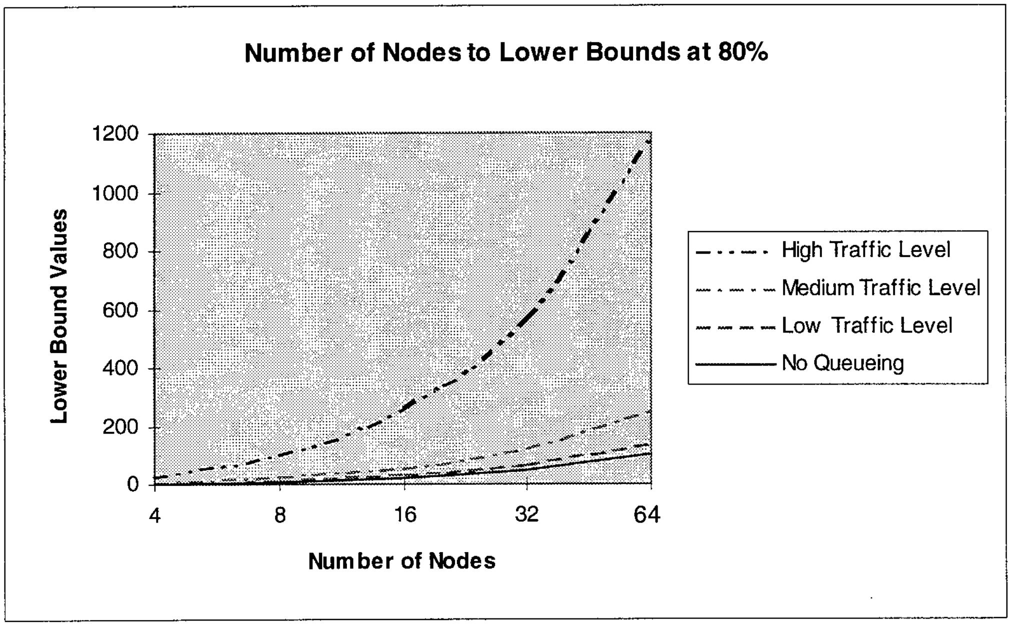


Figure 4.16 Doubling the Number of Nodes Compared to the Rough Doubling of the Lower Bound

Table 4.14 and Figure 4.16 both inherently show the engineers and designer the doubling of the number of nodes for any traffic conditions will more the double the lower bound value while maintaining the same 80% bounding assurance. This implies there are no positive gains as the network grows larger as more nodes are added to the system. Appendix D.3 contains the calculated output of the Excel spreadsheet program used.

Despite the diminishing gains as the number of nodes grow, new technologies from engineers can benefit the designers. This benefit comes from the increased assurance of the lower bound bounding the Total Sojourn Time. If the growth of the average delay per node can be kept constant as the number of nodes increases, the lower bound value can maintain the same growth rate and probability of bound will increase.

With the assumption of the new networking technology, Figures 4.17 through 4.19 show the extrapolation of the Total Delay Time and Lower Bound Value at a constant growth rate normalized to the four node data set from Network D. The probability of the lower bound properly bounding the Total Delay Time nears 100%. From the previous paragraphs, a doubling of the number of nodes at least doubles the total time. Given this model, engineers have a significant amount of time to reduce to reach a constant growth rate. Appendix D.4 through D.6 contains the Excel spreadsheet output for the constant growth calculations.

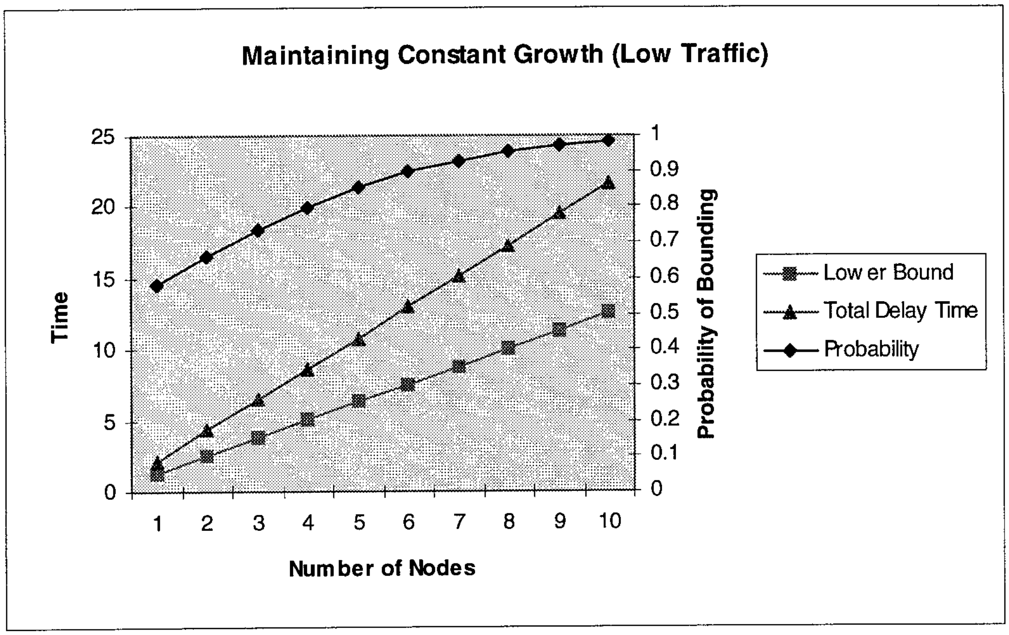


Figure 4.17 Constant Growth Rate (Low Traffic)

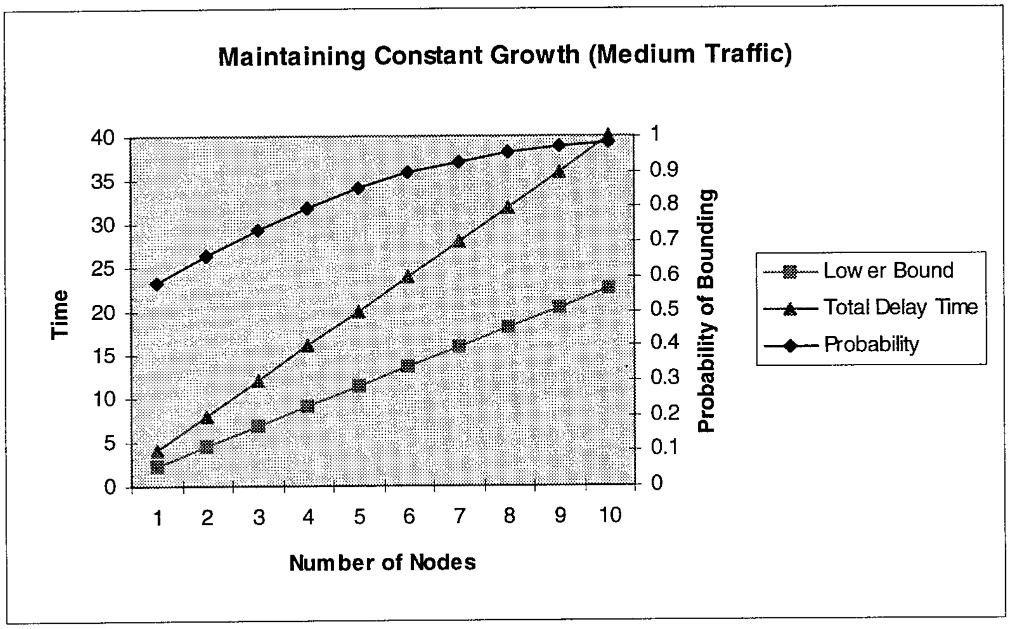


Figure 4.18 Constant Growth rate (Medium Traffic)

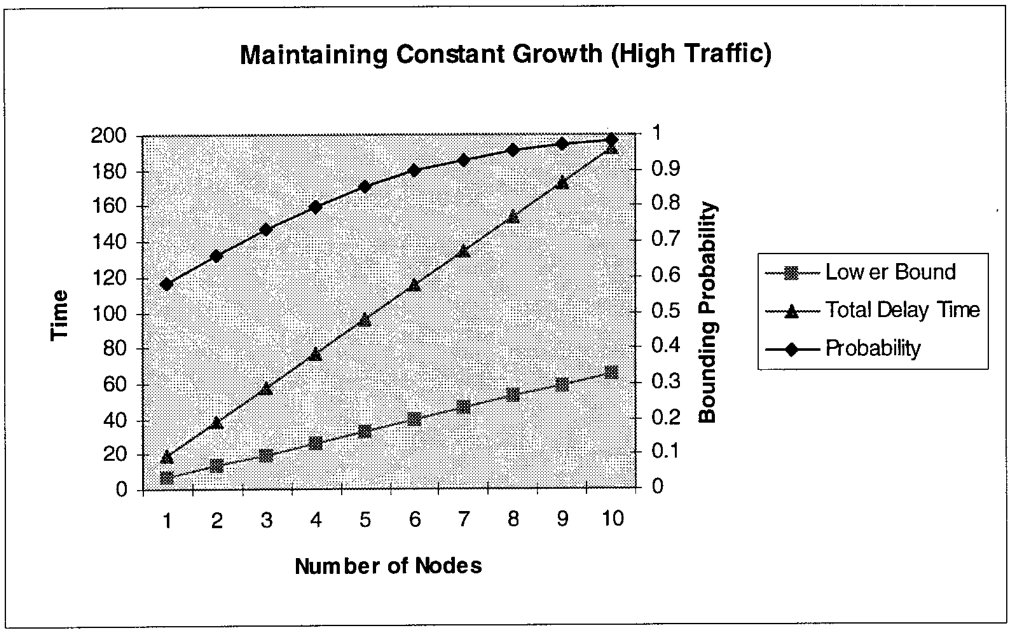


Figure 4.19 Constant growth Rate (High Traffic)

In addition to the lower bound estimation, the upper bound estimation technique, Equation (4.20) from Section 4.7, can also be applied to Network D. Network D satisfies the assumption (each node being an M/M/1 queue) required for using the second upper bound technique. Appendix D.7 contains the output of the Excel spreadsheet program showing the calculated upper bound values per node and their summation for a percentile value, $q$ , of 90. Table 4.15 summarizes these calculations.

Table 4.15 Upper Bound Values at 90 Percentile

| Node | Upper Bound (Low) | Upper Bound (Medium) | Upper Bound (High) |
|---|---|---|---|
| 1 | 3.8122 | 13.5149 | 30.5486 |
| 2 | 4.5205 | 14.8892 | 30.5476 |
| 3 | 5.0039 | 21.7876 | 228.2174 |
| 4 | 2.8881 | 8.3633 | 11.1985 |
| Total | 16.2247 | 58.5550 | 300.5111 |

Table 4.15 shows the upper bound of the entire High intensity primary route as 300.5111. This indicates to the user 90% of the sojourn times in Network D will not exceed the upper bound value. Referring to the representative trials in Network D from Table 4.11, the High intensity trial yields a Total Sojourn Time of 77.0190, which is appropriately within the upper bound value.

Figure 4.20 graphs both the lower and upper values applied to the three representative traffic levels in Network D. The displayed traffic levels generated by the GNA tool only represents the mean Total Sojourn Times.

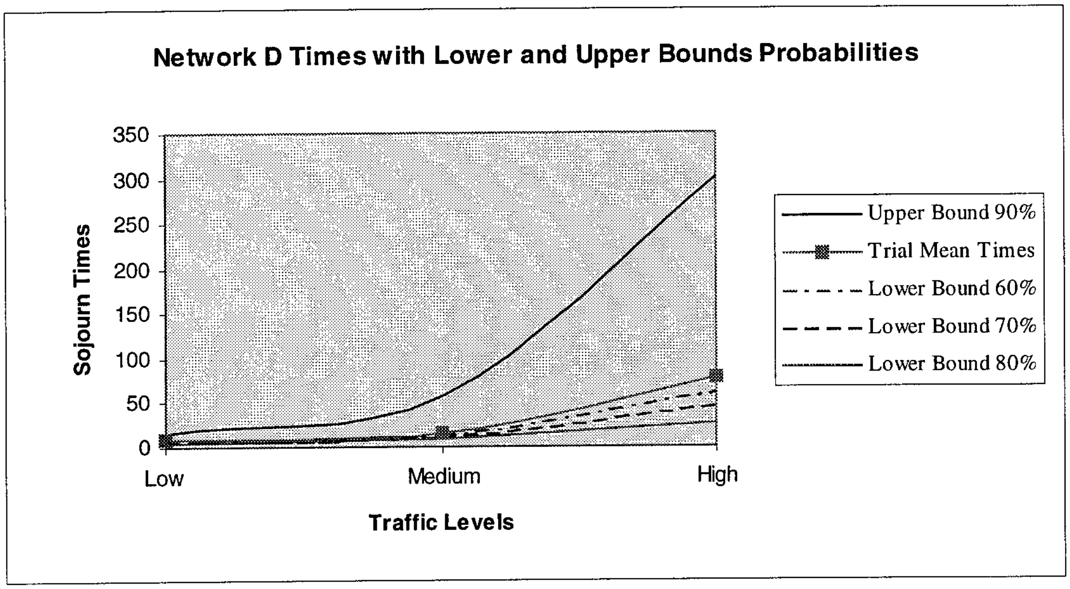


Figure 4.20 Network D Trial Times with Lower and Upper Bound Probabilities

From Figure 4.20, this demonstrate the two techniques can appropriately bound the various traffic loads used within Network D.

### *4.8 Summary*

In summary, this chapter analyzed the data created by the GNA tool. It showed specific link usage percentages as the Basic Secondary Traffic Rate and Primary Traffic Rate values changed. Usage Ratios showed the preference for the primary customer's path between different segments in the network. It showed choice of paths are drawn towards or pushed away from the nodes as their processing capabilities are altered. Additionally, this chapter provided a lower bound estimation of the sojourn time at a given probability for the path with independent and identically distributed nodes using the Central Limit Theorem. Two upper bound estimations were described. One provided the expected upper bound of sojourn time based on finding the maximum BSTR and PTR values which can be used in the GNA tool. The other estimation technique described the upper bound as a given percentile probability. The lower and upper bound techniques applied to a test network appropriately bounds the representative traffic levels at various probabilities. A discussion centered on the diminishing return aspect of the lower bound bounding probability as the number of nodes increased. Therefore, this GNA tool can be used in a variety of ways to produce data for analysis using the techniques described in this chapter.

THIS PAGE INTENTIONALLY LEFT BLANK.

# *V. CONCLUSIONS AND RECOMMENDATIONS*

This research has accomplished a number of objectives. The following sections discuss the conclusions of this research drawn for each objective introduced in Chapter 1.

## *5.1 Resource Constraints*

Upon review of the available development tools, the committee decided upon the use of the Generalized Network Analyzer (GNA) Tool. The models available within GNA were used to produce the Specific Link and Path Likelihood Prediction Tool. It was important that no additional development tools be used in order to maintain GNA as a self-contained package [ChR96].

## *5.2 Candidate Packet Switching Algorithm*

A number of packet switching routing algorithms were examined in Chapter Two of this study. Two journals provided key aspects to this tool. Alpert's discussion on the Prim-Dijkstra algorithm tradeoffs for routing indicated that Dijkstra's algorithm can effectively generate a route from source to destination as shown in Chapter Three [AlH95]. Cheng and Lin's work in joint virtual path assignment and virtual circuit routing provided the macro logic link aspect, called segments in Chapter Four, Tool Analysis, of this study, for analysis of path likelihood [ChL95].

### *5.3 GNA Constraints*

GNA is comprised of several analytical network models and conversion between them. The primary models are FORMULA and QNA. There are commonalties between the data representations needed for different models, and the GNA algorithms exploited them. Taking advantage of general and detailed models, one can appeal to different levels of decision making. Output has only to be analyzed and compared visually by inspecting graphical output on-screen and from print-screens. Symbolic representations of network components are used so that decision makers readily understand the results of analysis.

There exist a limitations within the GNA package. GNA's icon displays are preset internally to a fixed size and location. An option to resize the network is available. Since modifications to existing nodes will often occur, process capability and/or increased capacity labels will change. If resizing has occurred, the label may not appear in the intended location. It can appear away from the original node, over the node itself, or even over another node.

***5.3.1 FORMULA.*** FORMULA, the more general of the two models, was originally designed for investigating the behavior of stochastic communication networks. It is used to determine expected network performance (throughput and reliability), congestion points, and how throughput and network reliability can be improved by increasing the capacity and/or reliability of network components, based on a budget available for improvement.

***5.3.2 QNA.*** Queueing Network Analyzer (QNA), the more detailed model, was designed for the performance analysis of queueing networks. Given the Poisson arrival Process and exponentially distributed service times for M/M/1 queues, QNA calculates exact measures[1] of network congestion like expected queue lengths, throughput, traffic intensities, and expected sojourn time. Customer flows on the arcs are assumed to be random, and so are represented as stochastic processes. QNA has been used in this study to model the flow of electronic packets of information (commodity) through cables (routes) and computers (nodes). According to Whitt, QNA has produced good estimations[2] to simulations and other approximations (M/M/1, M/G/1, and GI/G/1) of queueing network performance [Whi83a].

The use of QNA within GNA for this study was not to create a simulation system. QNA cannot represent successive time intervals to generate distinct traffic messages. Specific individual messages cannot be identified for modification. Therefore, this packet switch study did not perform dynamic routing but concentrate on the virtual circuit aspect of routing.

### *5.4 Packet Switching Subsystem Implementation*

In this study, Dijkstra's algorithm has been modified to allow QNA's analysis output to act as a node's goodness metric. QNA's calculation of the expected Total Sojourn Time for the completion of queueing and service in a node provided accurate measurement[3] of expected congestion. Chapter Three verified and empirically validated the use of the modified Dijkstra's

[1] QNA formulas presented in Whitt's journal [Whi83a] were verified by hand with the aid of Excel to match the standard M/M/1 queueing formula from [Jai91].

[2] Whitt's work in [Whi83b] provides the performance of QNA by comparing it with simulations and other approximations of several networks of queues.

[3] QNA formulas presented in Whitt's journal [Whi83a] were verified by hand with the aid of Excel (same instance as Section 5.3.2) to match the standard M/M/1 queueing formula from [Jai91].

algorithm to properly deliver traffic and appropriately generate the lowest traffic path from a start node to a destination node.

### *5.5 Packet Switching Subsystem Integration*

The integration of the packet switching subsystem into GNA was successful[4]. Users can execute GNA to graphically develop a packet switching network. The network can be saved for later use or modification. The user can activate the Packet Switching Model Option from the GNA menu upon a packet switching network loaded from file. GNA will notify the user when certain network input parameters must be lowered or when a certain node must be improved. GNA displays the rejection notification and informs the user certain network input parameters must be lowered (PTR or BSTR) or where certain nodes must be improved to maintain stability. Without this instant unstable node identification provided by GNA, users cannot determine which node needs attention and improvements. Once this instability is removed, a good QoS is achieved. Upon completion of the analysis, GNA appends the generated route and expected sojourn time on to a text file for later analysis.

[4] The various Networks A through D were created and Total Sojourn Times calculated by using the GNA packet switching subsystem in GNA. Values reported by QNA's output file and those displayed by the subsystem were the same. Verified by visual inspection.

### *5.6 Packet Switching Subsystem Robustness*

The implementation of the subsystem includes the notification of input parameters exceeding the network's processing capability. Chapter Four of this study provided to the user three analytical techniques to estimate the bounds of the input parameters (One for the lower bound and two for the upper bound). Using the lower bound and the upper bound estimation techniques, Figure 4.20 showed Network D's sojourn times within 90% probability of the upper bound and 60% probability of the lower bound.

### *5.7 Summary of Objectives*

The listed objectives has focused this research to maintain GNA as a self contained package. The GNA environment allows the user to graphically create a network for analysis. The usefulness and accuracy of Whitt's QNA, shown in Chapter Three, makes it an excellent candidate for this study's prediction tool. Its integration as the core calculation tool provides the heart of the packet switching subsystem. QNA's output provides both the goodness metric of the nodes for Dijkstra's algorithm and the actual Total Sojourn Time calculations as explained in Chapter Three. This packet switching subsystem, using virtual circuit routing to deliver traffic from the user defined start node to the end node, produces results for user analysis. Using this GNA tool and concepts from Cheng and Lin's work [ChL95], the various networks analyzed in Chapter Four shows actual specific link usage percentage values. Chapter Four also presented and applied lower and upper bound estimation techniques to a test network. The lower and upper bound captured the test network's mean Total Sojourn Times at probabilities of 60% and 90%

respectively. A discussion centered on the diminishing return aspect of the lower bound bounding probability when the number of nodes are increased. This self contained GNA tool provides a working platform for further research.

### *5.8 Recommendations*

Although this research has only provided a basic tool to analyze specific link usage and path likelihood, additional aspects should be taken into account to provide a more realistic result. While this research attempted to present a representative amount of traffic conditions to demonstrate its use, the specific link usage is also subject to the varying traffic load levels. Generation of realistic traffic patterns is still a large area of research. New results from that research can be incorporated into this GNA tool.

The following recommendations are made in furthering the research of improving the performance of the GNA Specific Network Link and Path Likelihood Prediction Tool:

1. The code for the GNA Camera Option can be adjusted to reflect the new location of the nodes. The new location can be used by other procedures within GNA to appropriately place icon labels.

2. The GNA packet switching subsystem can be adjusted to repeat the QNA analysis of the network iteratively. Each iteration can represent a segment of time. During each segment, the QNA analysis provides the goodness metric of all the nodes in the network. Within this segment, the traffic is deliver from the source to destination along the route chosen by the routing algorithm. Upon the next segment, the QNA analysis will take into account the previous segment's traffic to determine a new goodness metric for all the nodes. Consequently, the routing algorithm will determine a new route to deliver the traffic. In effect, this can bring about a form of dynamic routing to this packet switching subsystem.

3. Once the tool integrates to ability to represent segments of time, link characteristics can be specified and the propagation delay value can be taken into account.

4. QNA's analysis can be used by routing algorithms as a goodness metric in a different application. The BONE'S Designer development tool may be able incorporate the QNA output as an estimation of node congestion.

### *5.9 Summary*

As the fast pace of warfare dictates the need for rapid information to assess battlefield conditions, any information that clarifies a battlefield situation is useful. The reliance of military commanders on network communications is vital to the operational status of their units. An accurate prediction of communication usage through a network will provide commanders with useful intelligence of friendly and unfriendly activities. Using the methodologies and techniques formulated in this research, this Specific Network Link and Path Likelihood Prediction Tool can give military commanders a basic working platform for additional intelligence information. Additionally, since QNA requires few calculations and GNA's Congestion Control provides unstable node identification, using the given bound estimation techniques, designers and engineers can evaluate network topologies much more easily.

THIS PAGE INTENTIONALLY LEFT BLANK

## ***Bibliography***

[AnN80] Y.P. Aneja, K.P.K. Nair, “Maximum Expected Flow in a Network Subject to Arc Failures.” *Network*, Vol. 10, pp. 45-47.

[AlH95] C.J. Alpert, T.C. Hu, J.H. Huang, A.B. Kahng, D. Karger, “Prim-Dijkstra Tradeoffs for Improved Performance-Driven Routing Tree Design.” *IEEE Transaction on Computer-Aided Design of Integrated Circuits and System*, Vol. 14, No. 7, pp. 890-896.

[All90] A. Allen, “Probability , Statistics, and Queueing Theory with Computer Science Applications Second Edition.” Academic Press Inc., San Diego, 1990.

[AwB90] B. Awerbuch, A. Baratz, D. Peleg. Cost-sensitive Analysis of Communication Protocols.” *Proceedings ACM Symposium: Principles Distributive Computing.* August 1990, pp. 177-187.

[Bal80] M.O. Ball, “Complexity of Network Reliability Components.” *Network* Vol. 10, pp.153-165.

[Chi93] A. A. Chien, “A Cost and Performance Model for K-ary N-cube Wormhole Adapters.” *Proceedings of Hot Onterconnecs Workshop,* August 1993.

[ChL95] K. T. Cheng, F. Y. S. Lin, “On the Joint Virtual Path Assignment and Virtual Circuit Routing Problem.” *1994 IEEE GLOBECOM. Communications: The Global Bridge. Conference Record,* 1995, pp. 777-782.

[ChR96] Y. Chan, Professor of Operations Research, R. Raines, Capt., Assistant Professor of Electrical Engineering, Air Force Institute of Technology, Air Education and Training Command, Wright-Patterson AFB OH. Personal interview. 18 Mar 1996.

[CoL90] T. H. Cormen, C. E. Leiserson, R. L. Rivest, “Introduction to Algorithms.” MIT Press, Massachusetts, 1990.

[CoK92] J. Cong, A.B. Kahng, G. Robins, M. Sarrafzadeh, C.K. Wong. "Provably Good Performance-Driven Global Routing." *IEEE Transaction: Computer-Aided Design*, Vol. 11, No 6, pp. 606-611.

[DaN94] J. G. Dai, V. Nguyen, M. I. Reiman, "Sequential Bottleneck Decomposition: An Approximation Method for Generalized Jackson Networks." *Operations Research*, Vol. 42, 1994, pp. 109-136.

[Dei96] D. Deitz, LtCol., Professor of Operations Research, Air Force Institute of Technology, Air Education and Training Command, Wright-Patterson AFB OH. Personal interview. 15 Oct. 1996.

[Dep84] Department of the Army. *Staff Organization and Operations.* FM 101-5. Washington: HQ DA, 1984.

[Dep85] Department of the Army. *Operational Terms and Symbols.* FM 101-5-1. Washington: HQ DA, October 1985.

[Dep93] Department of the Army. *Operations.* FM 100-5. Washington: HQ DA, June 1993.

[GeM80] E. Gelenbe, I. Mitrani, "Analysis and Synthesis of Computer Systems." Academic Press Inc., London, 1980.

[HiL94] J. L. Higle, W. Lowe, R. Odio, "Conditional Stochastic Decomposition: An Algorithmic Interface for Optimization and Simulation." *Operations Research*, Vol. 42, 1994, pp. 311-322.

[Jai91] R. Jain, "The Art of Computer Systems Performance Analysis," Wiley, New York, 1991.

[JeB80] P. Jensen, J.W. Barnes, "Network Flow Programming." Wiley, New York, 1980.

[KhR93] S. Khuller, B. Raghavachari, N. Young, "Balancing Minimum Spanning and Shortest Path Trees." *Proceedings ACM/SIAM Symposium: Discrete Algorithms*, January 1993, pp. 243-250.

[Kle75] L. Kleinrock, "Queueing Systems Volume I: Theory ." Wiley, New York, 1975.

[Klo96] J.M. Kloeber, LtCol. Class Handout, OPER 561, Military Simulation Systems, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, Spring Quarter 1996.

[Kue79] P. J. Kuehn, "Approximate Analysis of General Queueing Networks by Decomposition." *IEEE Trans. Commun.* Vol. 27, No. 1, 1979, pp. 113-26.

[LeY94] M.J. Lee, J.R. Yee, "An Algorithm for Optimal Minimax Routing in ATM Networks." *Annals of Operations Research.* Vol. 49, pp. 185-206.

[LiC94] Z. Liu, A. Chien, "Hierarchical Adaptive Routing." *Proceedings, Sixth IEEE Symposium on Parrallel and Dostributed Processing,* 1994, pp. 688-95.

[LiW93a] F. Y. S. Lin, J. L. Wang, "Minimax Open Shortest Path First Routing Algorithms in Networks Supporting the SMDS Service." *ICC '93 Geneva. IEEE International Conference on Communications '93. Technical Program, Conference Record,* 1993, pp. 666-670.

[LiW93b] F. Y. S. Lin, J. L. Wang, "A Minimax Utilization Routing Algorithm in Networks with Single Path Routing." *IEEE Global Telecommunications Conference, including a Communications Theory Mini Conference.. Technical Program Conference Record,* 1993, pp. 1067-1071.

[Mel86] B. Melamed, "The Performance Analysis Workstation: An Interactive Animated Simulation Package for Queueing Networks" *Proceedings of the 1986 Fall Joint Computer Conference,* pp. 729-737.

[MoE78] J.J. Moder, S.E. Elmaghraby, "Handbook of Operations Research: Foundations and Fundamentals." Van Nostrand Reinhold, New York, 1978.

[MuN94] M. Munafo, F. Neri, C. Scarpati, V.Vasco, "Analysis of the Spanning Tree and Source Routing LAN Interconnection Schemes." *Internetworking: research and Experience,* Vol. 5, pp. 121-149.

[Rai96a] R. A. Raines, Capt. Class Lecture, CSCE 654, Computer Communication Networks, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, Winter Quarter 1996.

[Rai96b] R. A. Raines, Capt. Class Handout, CSCE 754, Advanced Topics in Computer Communication Networks, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, Fall Quarter 1996.

[RoP91] D. F. Rogers, R. D. Plante, R. T. Wong, J. R. Evans, "Aggregation and Disaggregation Techniques and Methodology in Optimization." *Operations Research*, 394, 1991, pp. 553-582.

[SaA94] T. N. Saadawi, M. H. Ammar, A. E. Hakeem, "Fundamentals of Telecommunication Networks." John Wiley & Sons, New York, 1994.

[SaM95] D. Saha, A. Mukherjee, "Computational Analysis of an Optimal Routing Strategy in a Hierarchical Computer Communication Network." *Computer Communications* Vol. 18, No. 7, pp.507-511.

[See84] A. T. Seely, "User's Manual for GNA 1.0." AT&T Bell Laboratories, 1984.

[Spi75] M. R. Spiegel, "Schaum's Outline Series Theory and Problems of Probability and Statistics." McGraw-Hill Inc., 1975.

[TuA93] R. L. Turcotte, S. Arneson, S., "The Role of Commercial LEOs in Global Grid." *MILCOM 93: 1993 IEEE Military Communications Conference*, October 1993, pp. 247-251.

[Whi82] W. Whitt, "Approximating a Point Process by Renewal Process I: Two basic Methods", *Operations Research*, Vol. 30, No. 1 , 1982, pp. 125-47.

[Whi83a] W. Whitt, "Performance of the Queueing Network Analyzer", *The Bell System Technical Journal*, Vol. 62, No. 9 , 1983, pp. 2817-2843.

[Whi83b] W. Whitt, "The Queueing Network Analyzer", *The Bell System Technical Journal*, Vol. 62, No. 9 , 1983, pp. 2779-2815.

[Yim88] E. Yim, "Improving the Survivability of a Stochastic Communication Network" MS thesis, Department of Operations Research, U.S. Air Force Institute of Technology, OH, 1988.

## *Appendix A: Generated Route and Time for Network A*

```
PTR 5.00 PTRVar 1.00 BSTR 3.00 BSTRVar 1.00.
1 2 3
Expected sojourn time is 1.51950.
```

## *Appendix B.1: Total Sojourn Time Calculations for Network B*

| Node ID | 1 | 2 | 3 | 4 | 5 | Routes |
|---|---|---|---|---|---|---|
| m=servers | 40 | 20 | 25 | 40 | 40 | |
| L=arrival rate | 9 | 7 | 7 | 7 | 9 | |
| u=service rate | 3.33333E-01 | 5.00000E-01 | 2.50000E-01 | 1.00000E+00 | 5.00000E-01 | |
| r=L/mu=Intensity | 0.675 | 0.7 | 1.12 | 0.175 | 0.45 | |
| P[n=zero] | 1.87691E-12 | 8.16171E-07 | -2.29724E-12 | 9.11882E-04 | 1.57473E-08 | Route A=1-2-5 |
| P[Queueing] | 0.012719346 | 0.093561243 | 1.863507284 | 8.62504E-18 | 5.70293E-06 | 7.03412E+00 |
| E[N in system] | 2.70264E+01 | 1.42183E+01 | 1.06073E+01 | 7.00000E+00 | 1.80000E+01 | Route B=1-3-5 |
| E[N in queue] | 0.026417103 | 0.218309568 | -17.39273465 | 1.82955E-18 | 4.66603E-06 | 6.51826E+00 |
| E[waiting time] | 0.002935234 | 0.031187081 | -2.484676379 | 2.61365E-19 | 5.18448E-07 | Route C=1-4-5 |
| E[sojourn time] | 3.00294E+00 | 2.03119E+00 | 1.51532E+00 | 1.00000E+00 | 2.00000E+00 | 6.00294E+00 |
| (mr)^m | 1.79701E+57 | 8.36683E+22 | 1.50991E+36 | 6.36681E+33 | 1.62518E+50 | |
| m!*(1-r) | 2.65172E+47 | 7.29871E+17 | -1.86135E+24 | 6.7313E+47 | 4.48753E+47 | |
| Sum formula 1 to m-1 | 5.26015E+11 | 1110598.115 | 3.75888E+11 | 1095.633158 | 63502695.68 | |
| 1 | 27 | 14 | 28 | 7 | 18 | |
| 2 | 364.5 | 98 | 392 | 24.5 | 162 | |
| 3 | 3280.5 | 457.3333333 | 3658.666667 | 57.16666667 | 972 | |
| 4 | 22143.375 | 1600.666667 | 25610.66667 | 100.0416667 | 4374 | |
| 5 | 119574.225 | 4481.866667 | 143419.7333 | 140.0583333 | 15746.4 | |
| 6 | 538084.0125 | 10457.68889 | 669292.0889 | 163.4013889 | 47239.2 | |
| 7 | 2075466.905 | 20915.37778 | 2677168.356 | 163.4013889 | 121472.2286 | |
| 8 | 7004700.806 | 36601.91111 | 9370089.244 | 142.9762153 | 273312.5143 | |
| 9 | 21014102.42 | 56936.30617 | 29151388.76 | 111.203723 | 546625.0286 | |
| 10 | 56738076.53 | 79710.82864 | 81623888.53 | 77.8426061 | 983925.0514 | |
| 11 | 139266187.8 | 101450.1455 | 207769898.1 | 49.53620388 | 1610059.175 | |
| 12 | 313348922.6 | 118358.5031 | 484796428.8 | 28.89611893 | 2415088.763 | |
| 13 | 650801608.5 | 127463.0034 | 1044176924 | 15.55944865 | 3343969.056 | |
| 14 | 1255117388 | 127463.0034 | 2088353847 | 7.779724327 | 4299388.786 | |
| 15 | 2259211298 | 118965.4698 | 3898260515 | 3.630538019 | 5159266.543 | |
| 16 | 3812419066 | 104094.7861 | 6821955901 | 1.588360383 | 5804174.861 | |
| 17 | 6055018516 | 85725.11796 | 11236162661 | 0.654030746 | 6145596.912 | |
| 18 | 9082527774 | 66675.09174 | 17478475250 | 0.25434529 | 6145596.912 | |
| 19 | 12906749995 | 49129.01497 | 25757753000 | 0.09370616 | 5822144.443 | |
| 20 | 17424112493 | 34390.31048 | 36060854200 | 0.032797156 | 5239929.999 | |
| 21 | 22402430348 | 22926.87365 | 48081138934 | 0.010932385 | 4491368.57 | |
| 22 | 27493891791 | 14589.82869 | 61194176825 | 0.003478486 | 3674756.103 | |
| 23 | 32275438190 | 8880.765288 | 74497258743 | 0.00105867 | 2875896.081 | |
| 24 | 36309867963 | 5180.446418 | 86913468534 | 0.000308779 | 1.61174E-24 | |
| 25 | 39214657400 | 2901.049994 | 97343084758 | 8.6458E-05 | 1552983.884 | |
| 26 | 40722913454 | 1562.103843 | 1.04831E+11 | 2.32772E-05 | 1075142.689 | |
| 27 | 40722913454 | 809.9797705 | 1.08714E+11 | 6.03482E-06 | 716761.7924 | |
| 28 | 39268523688 | 404.9898852 | 1.08714E+11 | 1.5087E-06 | 460775.438 | |
| 29 | 36560349640 | 195.5123584 | 1.04965E+11 | 3.6417E-07 | 285998.5477 | |
| 30 | 32904314676 | 91.23910058 | 97967238277 | 8.4973E-08 | 171599.1286 | |
| 31 | 28658596654 | 41.2047551 | 88486537798 | 1.91875E-08 | 99638.20372 | |
| 32 | 24180690927 | 18.02708036 | 77425720574 | 4.19726E-09 | 56046.48959 | |
| 33 | 19784201667 | 7.647852272 | 65694550790 | 8.90327E-10 | 30570.8125 | |
| 34 | 15710983677 | 3.149115642 | 54101394768 | 1.83303E-10 | 16184.5478 | |
| 35 | 12119901694 | 1.259646257 | 43281115814 | 3.66605E-11 | 8323.481724 | |
| 36 | 9089926270 | 0.489862433 | 33663090078 | 7.12844E-12 | 4161.740862 | |
| 37 | 6633189440 | 0.185353353 | 25474770870 | 1.34862E-12 | 2024.63069 | |
| 38 | 4713055655 | 0.068288077 | 18770883799 | 2.48431E-13 | 959.0355898 | |
| 39 | 3262884684 | 0.024513669 | 13476531958 | 4.45901E-14 | 442.6318107 | |
| 40 | 2202447162 | 0.008579784 | 9433572371 | 7.80327E-15 | 199.1843148 | |

### *Appendix B.2: QNA Congestion Calculations for Network B*

```
1









     *****************************
      *              *
      *              *
      *   QNA 1.0  PROGRAM    *
      *   ----------------    *
      *              *
      *              *
      *  OUTPUT FOR  1  NETWORK   *
      *              *
      *              *
     *****************************




1INPUT DATA FOR NETWORK NO.    1
*********************************

THE INPUT DATA ARE:  SERVICE DATA AT EACH NODE PLUS ARRIVAL
           DATA BY ROUTES.



NUMBER OF NODES =    5


NUMBER OF SERVERS.....
```

```
         40      20      25      40      40


MEAN SERVICE TIMES.....
      0.30000D+01  0.20000D+01  0.40000D+01  0.10000D+01  0.20000D+01


SERVICE RATES.....
      0.33333D+00  0.50000D+00  0.25000D+00  0.10000D+01  0.50000D+00




***NOTE - THE USER HAS PROVIDED THE FOLLOWING VALUES FOR THE C-SQUARED OF
       THE SERVICE TIMES.




C-SQUARED FOR SERVICE TIMES.....
      0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01


VARIANCE OF SERVICE TIMES.....
      0.90000D+01  0.40000D+01  0.16000D+02  0.10000D+01  0.40000D+01




THE TOTAL NUMBER OF ROUTES IS     12


***NOTE - THE USER HAS PROVIDED THE C-SQUARED VALUES FOR THE ROUTES.


ROUTE VECTORS.....

             C-SQUARED    NO. OF
   FLOW RATE      OF FLOW     NODES     NODES ON ROUTE

  0.20000D+01  0.10000D+01     1      1

  0.20000D+01  0.10000D+01     1      2

  0.20000D+01  0.10000D+01     1      1

  0.20000D+01  0.10000D+01     1      4

  0.20000D+01  0.10000D+01     1      1
```

```
 0.20000D+01   0.10000D+01     1     3

 0.20000D+01   0.10000D+01     1     2

 0.20000D+01   0.10000D+01     1     5

 0.20000D+01   0.10000D+01     1     4

 0.20000D+01   0.10000D+01     1     5

 0.20000D+01   0.10000D+01     1     3

 0.20000D+01   0.10000D+01     1     5




ROUTING MATRIX.....

  ROW  Q(K,J)

    1  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


EXTERNAL ARRIVAL RATES.....
       0.60000D+01  0.40000D+01  0.40000D+01  0.40000D+01  0.60000D+01


MEAN INTERARRIVAL TIMES.....
       0.16667D+00  0.25000D+00  0.25000D+00  0.25000D+00  0.16667D+00



RESULTS OF INPUT ERROR CHECK.
******************************
```

```
  NO INPUT ERRORS WERE DETECTED.



FEEDBACK CALCULATIONS.
**********************


USER HAS REQUESTED REMOVAL OF IMMEDIATE FEEDBACK, IF ANY.

THERE WAS NO IMMEDIATE FEEDBACK.




RESULTS COMPUTED BY MAIN PROGRAM, PART 1.
SOLVING THE TRAFFIC RATE EQUATIONS.
****************************************


ARRIVAL RATE TO EACH NODE.....
     0.60000D+01  0.40000D+01  0.40000D+01  0.40000D+01  0.60000D+01


TRAFFIC INTENSITY AT EACH NODE.....
     0.45000D+00  0.40000D+00  0.64000D+00  0.10000D+00  0.30000D+00


ARRIVAL RATE ON ARC TO "J" FROM "K".....

  ROW  ARCARV(K,J)

   1  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


PROPORTION OF ARRIVALS TO "J" THAT CAME FROM "K".....

  ROW  PROPAR(K,J)
  1  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00

  2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00

  3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00

  4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00

  5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


DEPARTURE RATE FROM THE NETWORK AT EACH NODE.....
     0.60000D+01  0.40000D+01  0.40000D+01  0.40000D+01  0.60000D+01


TOTAL FLOWS.....
   FLOW OUT OF SYSTEM =  0.24000000D+02
   FLOW  INTO  SYSTEM =  0.24000000D+02


********************************************************************************


***NOTE - THE PROGRAM HAS COMPUTED THE FOLLOWING C-SQUARED VALUES.


C-SQUARED FOR EXTERNAL ARRIVAL PROCESSES.....
     0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01




RESULTS COMPUTED BY MAIN PROGRAM, PART 2.
SOLVING THE TRAFFIC VARIABILITY EQUATIONS.
*******************************************


PARAMETERS FOR THE LINEAR EQUATIONS.....

  X(I).....
     0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01

  W(J).....
     0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01

  ALPHA(J).....
     0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01
  ROW  BETA(I,J)

   1  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


C-SQUARED FOR ARRIVAL PROCESSES (CSQARV(J)).....
      0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01

C-SQUARED FOR DEPARTURE PROCESSES (CSQDEP(J)).....
      0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01

*********************************************************************************




NETWORK    1
GI/G/N QUEUE AT NODE       1
----------------------------



INPUT.....
  ARRIVAL RATE          =  0.6000000D+01
  MEAN INTERARRIVAL TIME =  0.1666667D+00
  SERVICE RATE          =  0.3333333D+00
  MEAN SERVICE TIME     =  0.3000000D+01
  TRAFFIC INTENSITY     =  0.4500000D+00
  C-SQUARED-A           =  0.1000000D+01
  C-SQUARED-S           =  0.1000000D+01
  NUMBER OF SERVERS     =          40


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY =  0.5469786D-05
  MEAN NUMBER OF BUSY SERVERS      =  0.1800000D+02
  EW    , MEAN DELAY          = 0.7458799D-06
  E(W**2), SECOND MOMENT      = 0.5563369D-12
  VARIANCE OF W               = 0.0000000D+00
  C-SQUARED OF W              = 0.0000000D+00

  EN    , MEAN NO. IN SYSTEM = 0.1800000D+02
  E(N**2), SECOND MOMENT     = 0.3420003D+03
  VARIANCE OF N              = 0.1800011D+02
  C-SQUARED OF N             = 0.5555587D-01




NETWORK    1
GI/G/N QUEUE AT NODE      2
----------------------------



INPUT.....
  ARRIVAL RATE           = 0.4000000D+01
  MEAN INTERARRIVAL TIME = 0.2500000D+00
  SERVICE RATE           = 0.5000000D+00
  MEAN SERVICE TIME      = 0.2000000D+01
  TRAFFIC INTENSITY      = 0.4000000D+00
  C-SQUARED-A            = 0.1000000D+01
  C-SQUARED-S            = 0.1000000D+01
  NUMBER OF SERVERS      =          20


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.2648799D-03
  MEAN NUMBER OF BUSY SERVERS      = 0.8000000D+01

  EW    , MEAN DELAY          = 0.4414665D-04
  E(W**2), SECOND MOMENT      = 0.1471555D-04
  VARIANCE OF W               = 0.1471360D-04
  C-SQUARED OF W              = 0.7549592D+04

  EN    , MEAN NO. IN SYSTEM = 0.8000177D+01
  E(N**2), SECOND MOMENT     = 0.7200536D+02
  VARIANCE OF N              = 0.8002531D+01
  C-SQUARED OF N             = 0.1250340D+00




NETWORK    1
```

```
GI/G/N QUEUE AT NODE      3
----------------------------



INPUT.....
  ARRIVAL RATE          = 0.4000000D+01
  MEAN INTERARRIVAL TIME = 0.2500000D+00
  SERVICE RATE          = 0.2500000D+00
  MEAN SERVICE TIME     = 0.4000000D+01
  TRAFFIC INTENSITY     = 0.6400000D+00
  C-SQUARED-A           = 0.1000000D+01
  C-SQUARED-S           = 0.1000000D+01
  NUMBER OF SERVERS     =          25


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.2546456D-01
  MEAN NUMBER OF BUSY SERVERS      = 0.1600000D+02

  EW     , MEAN DELAY        = 0.1131758D-01
  E(W**2), SECOND MOMENT     = 0.1006007D-01
  VARIANCE OF W              = 0.9931984D-02
  C-SQUARED OF W             = 0.7754054D+02

  EN     , MEAN NO. IN SYSTEM = 0.1604527D+02
  E(N**2), SECOND MOMENT      = 0.2740623D+03
  VARIANCE OF N               = 0.1661161D+02
  C-SQUARED OF N              = 0.6452348D-01




NETWORK    1
GI/G/N QUEUE AT NODE      4
----------------------------



INPUT.....
  ARRIVAL RATE          = 0.4000000D+01
  MEAN INTERARRIVAL TIME = 0.2500000D+00
  SERVICE RATE          = 0.1000000D+01
  MEAN SERVICE TIME     = 0.1000000D+01
  TRAFFIC INTENSITY     = 0.1000000D+00
  C-SQUARED-A           = 0.1000000D+01
  C-SQUARED-S           = 0.1000000D+01
  NUMBER OF SERVERS     =          40
```

```
RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY =  0.0000000D+00
  MEAN NUMBER OF BUSY SERVERS      =  0.4000000D+01

  EW     , MEAN DELAY          =  0.0000000D+00
  E(W**2), SECOND MOMENT       =  0.0000000D+00
  VARIANCE OF W                =  0.0000000D+00
  C-SQUARED OF W               =  0.0000000D+00

  EN     , MEAN NO. IN SYSTEM =  0.4000000D+01
  E(N**2), SECOND MOMENT      =  0.2000000D+02
  VARIANCE OF N               =  0.4000000D+01
  C-SQUARED OF N              =  0.2500000D+00




NETWORK    1
GI/G/N QUEUE AT NODE       5
-----------------------------



INPUT.....
  ARRIVAL RATE             =  0.6000000D+01
  MEAN INTERARRIVAL TIME =  0.1666667D+00
  SERVICE RATE             =  0.5000000D+00
  MEAN SERVICE TIME       =  0.2000000D+01
  TRAFFIC INTENSITY       =  0.3000000D+00
  C-SQUARED-A             =  0.1000000D+01
  C-SQUARED-S             =  0.1000000D+01
  NUMBER OF SERVERS       =          40


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY =  0.0000000D+00
  MEAN NUMBER OF BUSY SERVERS      =  0.1200000D+02

  EW     , MEAN DELAY          =  0.0000000D+00
  E(W**2), SECOND MOMENT       =  0.0000000D+00
  VARIANCE OF W                =  0.0000000D+00
  C-SQUARED OF W               =  0.0000000D+00

  EN     , MEAN NO. IN SYSTEM =  0.1200000D+02
  E(N**2), SECOND MOMENT      =  0.1560000D+03
  VARIANCE OF N               =  0.1200000D+02
  C-SQUARED OF N              =  0.8333333D-01
```

```
*******************************************************************************


TOTAL NETWORK PERFORMANCE FOR NETWORK    1


THROUGHPUT =   0.24000D+02



EXPECTED NUMBER OF VISITS TO EACH NODE.....
     0.25000D+00  0.16667D+00  0.16667D+00  0.16667D+00  0.25000D+00


EXPECTED SOJOURN TIME PER VISIT IN EACH NODE.....
     0.30000D+01  0.20000D+01  0.40113D+01  0.10000D+01  0.20000D+01


VARIANCE OF THE SOJOURN TIME PER VISIT IN EACH NODE.....
     0.90000D+01  0.40000D+01  0.16010D+02  0.10000D+01  0.40000D+01


EXPECTED TOTAL SOJOURN TIME IN EACH NODE.....
     0.75000D+00  0.33334D+00  0.66855D+00  0.16667D+00  0.50000D+00


EXPECTED TOTAL SOJOURN TIME IN THE NETWORK =   0.24186D+01



EXPECTED NUMBER IN EACH NODE.....
     0.18000D+02  0.80002D+01  0.16045D+02  0.40000D+01  0.12000D+02


EXPECTED NUMBER IN NETWORK =    0.58045D+02


VARIANCE OF NUMBER IN EACH NODE.....
     0.18000D+02  0.80025D+01  0.16612D+02  0.40000D+01  0.12000D+02


VARIANCE OF NUMBER IN NETWORK =    0.58614D+02

*******************************************************************************
```

```
SOJOURN TIMES BY ROUTES FOR NETWORK   1

                MEAN TOTAL      MEAN TOTAL     MEAN TOTAL
               SERVICE TIME       DELAY       SOJOURN TIME
   ROUTE NO.     ON ROUTE        ON ROUTE       ON ROUTE

      1       0.30000D+01    0.74588D-06    0.30000D+01

      2       0.20000D+01    0.44147D-04    0.20000D+01

      3       0.30000D+01    0.74588D-06    0.30000D+01

      4       0.10000D+01    0.00000D+00    0.10000D+01

      5       0.30000D+01    0.74588D-06    0.30000D+01

      6       0.40000D+01    0.11318D-01    0.40113D+01

      7       0.20000D+01    0.44147D-04    0.20000D+01

      8       0.20000D+01    0.00000D+00    0.20000D+01

      9       0.10000D+01    0.00000D+00    0.10000D+01

     10       0.20000D+01    0.00000D+00    0.20000D+01

     11       0.40000D+01    0.11318D-01    0.40113D+01

     12       0.20000D+01    0.00000D+00    0.20000D+01

               VARIANCE OF     VARIANCE OF    VARIANCE OF
               SERVICE TIME       DELAY       SOJOURN TIME
   ROUTE NO.     ON ROUTE        ON ROUTE       ON ROUTE

      1       0.90000D+01    0.00000D+00    0.90000D+01

      2       0.40000D+01    0.14714D-04    0.40000D+01

      3       0.90000D+01    0.00000D+00    0.90000D+01

      4       0.10000D+01    0.00000D+00    0.10000D+01

      5       0.90000D+01    0.00000D+00    0.90000D+01

      6       0.16000D+02    0.99320D-02    0.16010D+02

      7       0.40000D+01    0.14714D-04    0.40000D+01

      8       0.40000D+01    0.00000D+00    0.40000D+01

      9       0.10000D+01    0.00000D+00    0.10000D+01
```

```
10     0.40000D+01    0.00000D+00    0.40000D+01

11     0.16000D+02    0.99320D-02    0.16010D+02

12     0.40000D+01    0.00000D+00    0.40000D+01

****************************************************************************************
```

## *Appendix B.3: QNA Total Sojourn Time Calculations for Network B*

```
1
```

```
*****************************
 *                  *
 *                  *
 *    QNA 1.0  PROGRAM     *
 *    ---------------     *
 *                  *
 *                  *
 *  OUTPUT FOR  1  NETWORK  *
 *                  *
 *                  *
*****************************
```

```
1INPUT DATA FOR NETWORK NO.    1
*******************************

THE INPUT DATA ARE:  SERVICE DATA AT EACH NODE PLUS ARRIVAL
              DATA BY ROUTES.



NUMBER OF NODES =     5


NUMBER OF SERVERS.....
```

```
        40     20     25     40     40


MEAN SERVICE TIMES.....
        0.30000D+01  0.20000D+01  0.40000D+01  0.10000D+01  0.20000D+01


SERVICE RATES.....
        0.33333D+00  0.50000D+00  0.25000D+00  0.10000D+01  0.50000D+00



***NOTE - THE USER HAS PROVIDED THE FOLLOWING VALUES FOR THE C-SQUARED OF
         THE SERVICE TIMES.



C-SQUARED FOR SERVICE TIMES.....
        0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01


VARIANCE OF SERVICE TIMES.....
        0.90000D+01  0.40000D+01  0.16000D+02  0.10000D+01  0.40000D+01



THE TOTAL NUMBER OF ROUTES IS    13


***NOTE - THE USER HAS PROVIDED THE C-SQUARED VALUES FOR THE ROUTES.


ROUTE VECTORS.....

                C-SQUARED    NO. OF
   FLOW RATE     OF FLOW     NODES     NODES ON ROUTE

  0.20000D+01   0.10000D+01      1      1

  0.20000D+01   0.10000D+01      1      2

  0.20000D+01   0.10000D+01      1      1

  0.20000D+01   0.10000D+01      1      4

  0.20000D+01   0.10000D+01      1      1
```

```
 0.20000D+01   0.10000D+01      1      3

 0.20000D+01   0.10000D+01      1      2

 0.20000D+01   0.10000D+01      1      5

 0.20000D+01   0.10000D+01      1      4

 0.20000D+01   0.10000D+01      1      5

 0.20000D+01   0.10000D+01      1      3

 0.20000D+01   0.10000D+01      1      5

 0.30000D+01   0.10000D+01      3      1    4    5




ROUTING MATRIX.....

 ROW  Q(K,J)

  1  0.00000D+00  0.00000D+00  0.00000D+00  0.33333D+00  0.00000D+00


  2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


  3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


  4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.42857D+00


  5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


EXTERNAL ARRIVAL RATES.....
     0.90000D+01  0.40000D+01  0.40000D+01  0.40000D+01  0.60000D+01


MEAN INTERARRIVAL TIMES.....
     0.11111D+00  0.25000D+00  0.25000D+00  0.25000D+00  0.16667D+00
```

```
RESULTS OF INPUT ERROR CHECK.
*****************************
 NO INPUT ERRORS WERE DETECTED.



FEEDBACK CALCULATIONS.
**********************


USER HAS REQUESTED REMOVAL OF IMMEDIATE FEEDBACK, IF ANY.

THERE WAS NO IMMEDIATE FEEDBACK.



RESULTS COMPUTED BY MAIN PROGRAM, PART 1.
SOLVING THE TRAFFIC RATE EQUATIONS.
*****************************************


ARRIVAL RATE TO EACH NODE.....
     0.90000D+01  0.40000D+01  0.40000D+01  0.70000D+01  0.60000D+01


TRAFFIC INTENSITY AT EACH NODE.....
     0.67500D+00  0.40000D+00  0.64000D+00  0.17500D+00  0.30000D+00


ARRIVAL RATE ON ARC TO "J" FROM "K".....

  ROW  ARCARV(K,J)

   1  0.00000D+00  0.30000D+01  0.00000D+00  0.00000D+00  0.00000D+00


   2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


PROPORTION OF ARRIVALS TO "J" THAT CAME FROM "K".....
```

```
  ROW  PROPAR(K,J)

    1  0.00000D+00  0.42857D+00  0.00000D+00  0.00000D+00  0.00000D+00


    2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


    5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


DEPARTURE RATE FROM THE NETWORK AT EACH NODE.....
      0.60000D+01  0.40000D+01  0.40000D+01  0.70000D+01  0.60000D+01


TOTAL FLOWS.....
    FLOW OUT OF SYSTEM =  0.27000000D+02
    FLOW  INTO  SYSTEM =  0.27000000D+02


********************************************************************************


***NOTE - THE PROGRAM HAS COMPUTED THE FOLLOWING C-SQUARED VALUES.


C-SQUARED FOR EXTERNAL ARRIVAL PROCESSES.....
      0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01




RESULTS COMPUTED BY MAIN PROGRAM, PART 2.
SOLVING THE TRAFFIC VARIABILITY EQUATIONS.
******************************************


PARAMETERS FOR THE LINEAR EQUATIONS.....

  X(I).....
      0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01  0.10000D+01

  W(J).....
      0.10000D+01  0.25189D+00  0.10000D+01  0.17315D+01  0.10000D+01

  ALPHA(J).....
```

```
     0.10000D+01  0.89205D+00  0.10000D+01  0.16074D+01  0.10000D+01

  ROW  BETA(I,J)

   1  0.00000D+00  0.13465D+00  0.00000D+00  0.00000D+00  0.00000D+00


   2  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   3  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   4  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


   5  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00  0.00000D+00


C-SQUARED FOR ARRIVAL PROCESSES (CSQARV(J)).....
     0.10000D+01  0.89205D+00  0.10000D+01  0.17421D+01  0.10000D+01

C-SQUARED FOR DEPARTURE PROCESSES (CSQDEP(J)).....
     0.10000D+01  0.90932D+00  0.10000D+01  0.17193D+01  0.10000D+01


***********************************************************************************




NETWORK    1
GI/G/N QUEUE AT NODE       1
----------------------------



INPUT.....
  ARRIVAL RATE          = 0.9000000D+01
  MEAN INTERARRIVAL TIME = 0.1111111D+00
  SERVICE RATE          = 0.3333333D+00
  MEAN SERVICE TIME      = 0.3000000D+01
  TRAFFIC INTENSITY      = 0.6750000D+00
  C-SQUARED-A           = 0.1000000D+01
  C-SQUARED-S           = 0.1000000D+01
  NUMBER OF SERVERS      =          40


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.1271920D-01
  MEAN NUMBER OF BUSY SERVERS      = 0.2700000D+02

  EW    , MEAN DELAY         = 0.2935200D-02
  E(W**2), SECOND MOMENT      = 0.1354708D-02
  VARIANCE OF W            = 0.1346092D-02
  C-SQUARED OF W            = 0.1562426D+03

  EN    , MEAN NO. IN SYSTEM = 0.2702642D+02
  E(N**2), SECOND MOMENT      = 0.7579061D+03
  VARIANCE OF N            = 0.2747887D+02
  C-SQUARED OF N            = 0.3762027D-01




NETWORK    1
GI/G/N QUEUE AT NODE      2
----------------------------



INPUT.....
  ARRIVAL RATE          = 0.4000000D+01
  MEAN INTERARRIVAL TIME = 0.2500000D+00
  SERVICE RATE          = 0.5000000D+00
  MEAN SERVICE TIME      = 0.2000000D+01
  TRAFFIC INTENSITY      = 0.4000000D+00
  C-SQUARED-A           = 0.8920475D+00
  C-SQUARED-S           = 0.1000000D+01
  NUMBER OF SERVERS      =          20


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.2648799D-03
  MEAN NUMBER OF BUSY SERVERS      = 0.8000000D+01

  EW    , MEAN DELAY         = 0.4176378D-04
  E(W**2), SECOND MOMENT      = 0.1316984D-04
  VARIANCE OF W            = 0.1316810D-04
  C-SQUARED OF W            = 0.7549592D+04

  EN    , MEAN NO. IN SYSTEM = 0.8000167D+01
  E(N**2), SECOND MOMENT      = 0.7200518D+02
  VARIANCE OF N            = 0.8002512D+01
  C-SQUARED OF N            = 0.1250340D+00
```

```
NETWORK    1
GI/G/N QUEUE AT NODE       3
----------------------------



INPUT.....
  ARRIVAL RATE           = 0.4000000D+01
  MEAN INTERARRIVAL TIME = 0.2500000D+00
  SERVICE RATE           = 0.2500000D+00
  MEAN SERVICE TIME      = 0.4000000D+01
  TRAFFIC INTENSITY      = 0.6400000D+00
  C-SQUARED-A            = 0.1000000D+01
  C-SQUARED-S            = 0.1000000D+01
  NUMBER OF SERVERS      =           25

RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.2546456D-01
  MEAN NUMBER OF BUSY SERVERS      = 0.1600000D+02

  EW     , MEAN DELAY        = 0.1131758D-01
  E(W**2), SECOND MOMENT     = 0.1006007D-01
  VARIANCE OF W              = 0.9931984D-02
  C-SQUARED OF W             = 0.7754054D+02

  EN     , MEAN NO. IN SYSTEM = 0.1604527D+02
  E(N**2), SECOND MOMENT      = 0.2740623D+03
  VARIANCE OF N               = 0.1661161D+02
  C-SQUARED OF N              = 0.6452348D-01




NETWORK    1
GI/G/N QUEUE AT NODE       4
----------------------------



INPUT.....
  ARRIVAL RATE           = 0.7000000D+01
  MEAN INTERARRIVAL TIME = 0.1428571D+00
  SERVICE RATE           = 0.1000000D+01
  MEAN SERVICE TIME      = 0.1000000D+01
  TRAFFIC INTENSITY      = 0.1750000D+00
  C-SQUARED-A            = 0.1742060D+01
  C-SQUARED-S            = 0.1000000D+01
  NUMBER OF SERVERS      =           40
```

```
RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.0000000D+00
  MEAN NUMBER OF BUSY SERVERS      = 0.7000000D+01

  EW     , MEAN DELAY          = 0.0000000D+00
  E(W**2), SECOND MOMENT       = 0.0000000D+00
  VARIANCE OF W                = 0.0000000D+00
  C-SQUARED OF W               = 0.0000000D+00

  EN     , MEAN NO. IN SYSTEM = 0.7000000D+01
  E(N**2), SECOND MOMENT       = 0.5600000D+02
  VARIANCE OF N                = 0.7000000D+01
  C-SQUARED OF N               = 0.1428571D+00




NETWORK    1
GI/G/N QUEUE AT NODE      5
----------------------------



INPUT.....
  ARRIVAL RATE           = 0.6000000D+01
  MEAN INTERARRIVAL TIME = 0.1666667D+00
  SERVICE RATE           = 0.5000000D+00
  MEAN SERVICE TIME      = 0.2000000D+01
  TRAFFIC INTENSITY      = 0.3000000D+00
  C-SQUARED-A            = 0.1000000D+01
  C-SQUARED-S            = 0.1000000D+01
  NUMBER OF SERVERS      =         40


RESULTS.....

  PROBABILITY ALL SERVERS ARE BUSY = 0.0000000D+00
  MEAN NUMBER OF BUSY SERVERS      = 0.1200000D+02

  EW     , MEAN DELAY          = 0.0000000D+00
  E(W**2), SECOND MOMENT       = 0.0000000D+00
  VARIANCE OF W                = 0.0000000D+00
  C-SQUARED OF W               = 0.0000000D+00

  EN     , MEAN NO. IN SYSTEM = 0.1200000D+02
  E(N**2), SECOND MOMENT       = 0.1560000D+03
  VARIANCE OF N                = 0.1200000D+02
  C-SQUARED OF N               = 0.8333333D-01
```

```
********************************************************************************

TOTAL NETWORK PERFORMANCE FOR NETWORK    1

THROUGHPUT =   0.27000D+02



EXPECTED NUMBER OF VISITS TO EACH NODE.....
         0.33333D+00   0.14815D+00   0.14815D+00   0.25926D+00   0.22222D+00

EXPECTED SOJOURN TIME PER VISIT IN EACH NODE.....
         0.30029D+01   0.20000D+01   0.40113D+01   0.10000D+01   0.20000D+01

VARIANCE OF THE SOJOURN TIME PER VISIT IN EACH NODE.....
         0.90013D+01   0.40000D+01   0.16010D+02   0.10000D+01   0.40000D+01

EXPECTED TOTAL SOJOURN TIME IN EACH NODE.....
         0.10010D+01   0.29630D+00   0.59427D+00   0.25926D+00   0.44444D+00

EXPECTED TOTAL SOJOURN TIME IN THE NETWORK =   0.25953D+01



EXPECTED NUMBER IN EACH NODE.....
         0.27026D+02   0.80002D+01   0.16045D+02   0.70000D+01   0.12000D+02

EXPECTED NUMBER IN NETWORK =     0.70072D+02

VARIANCE OF NUMBER IN EACH NODE.....
         0.27479D+02   0.80025D+01   0.16612D+02   0.70000D+01   0.12000D+02

VARIANCE OF NUMBER IN NETWORK =    0.71093D+02

********************************************************************************
```

SOJOURN TIMES BY ROUTES FOR NETWORK 1

|  | MEAN TOTAL | MEAN TOTAL | MEAN TOTAL |
|---|---|---|---|
|  | SERVICE TIME | DELAY | SOJOURN TIME |
| ROUTE NO. | ON ROUTE | ON ROUTE | ON ROUTE |
| 1 | 0.30000D+01 | 0.29352D-02 | 0.30029D+01 |
| 2 | 0.20000D+01 | 0.41764D-04 | 0.20000D+01 |
| 3 | 0.30000D+01 | 0.29352D-02 | 0.30029D+01 |
| 4 | 0.10000D+01 | 0.00000D+00 | 0.10000D+01 |
| 5 | 0.30000D+01 | 0.29352D-02 | 0.30029D+01 |
| 6 | 0.40000D+01 | 0.11318D-01 | 0.40113D+01 |
| 7 | 0.20000D+01 | 0.41764D-04 | 0.20000D+01 |
| 8 | 0.20000D+01 | 0.00000D+00 | 0.20000D+01 |
| 9 | 0.10000D+01 | 0.00000D+00 | 0.10000D+01 |
| 10 | 0.20000D+01 | 0.00000D+00 | 0.20000D+01 |
| 11 | 0.40000D+01 | 0.11318D-01 | 0.40113D+01 |
| 12 | 0.20000D+01 | 0.00000D+00 | 0.20000D+01 |
| 13 | 0.60000D+01 | 0.29352D-02 | 0.60029D+01 |

|  | VARIANCE OF | VARIANCE OF | VARIANCE OF |
|---|---|---|---|
|  | SERVICE TIME | DELAY | SOJOURN TIME |
| ROUTE NO. | ON ROUTE | ON ROUTE | ON ROUTE |
| 1 | 0.90000D+01 | 0.13461D-02 | 0.90013D+01 |
| 2 | 0.40000D+01 | 0.13168D-04 | 0.40000D+01 |
| 3 | 0.90000D+01 | 0.13461D-02 | 0.90013D+01 |
| 4 | 0.10000D+01 | 0.00000D+00 | 0.10000D+01 |
| 5 | 0.90000D+01 | 0.13461D-02 | 0.90013D+01 |
| 6 | 0.16000D+02 | 0.99320D-02 | 0.16010D+02 |
| 7 | 0.40000D+01 | 0.13168D-04 | 0.40000D+01 |

```
 8     0.40000D+01    0.00000D+00    0.40000D+01

 9     0.10000D+01    0.00000D+00    0.10000D+01

10     0.40000D+01    0.00000D+00    0.40000D+01

11     0.16000D+02    0.99320D-02    0.16010D+02

12     0.40000D+01    0.00000D+00    0.40000D+01

13     0.14000D+02    0.13461D-02    0.14001D+02

****************************************************************************************************
```

## *Appendix C.1: Specific Link Analysis Technique Calculations for Network C1*

| Network C1 | Factors | | Links | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Trial | PTR | BSTR | 1_15 | 1_16 | 15_4 | 16_3 | 4_20 | 3_19 | 20_23 | 19_23 | 23_18 | 18_9 | 18_11 | 9_25 | 11_25 | 25_14 |
| 1 | 1.3 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 0 | 1 | 0 | 1 |
| 2 | 1.3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 3 | 1.3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 4 | 1.4 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 0 | 1 | 0 | 1 |
| 5 | 1.4 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 6 | 1.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 7 | 1.5 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 0 | 1 | 0 | 1 |
| 8 | 1.5 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 9 | 1.5 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 10 | 1.6 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 0 | 1 | 0 | 1 |
| 11 | 1.6 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 12 | 1.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 13 | 1.7 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 14 | 1.7 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 15 | 1.8 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 16 | 1.8 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 17 | 1.9 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 18 | 1.9 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 19 | 2 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 20 | 2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 21 | 2.1 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 22 | 2.1 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 23 | 2.2 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 24 | 2.2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 25 | 2.3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 26 | 2.3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 27 | 2.4 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 28 | 2.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 29 | 2.6 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 30 | 2.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 31 | 2.8 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 32 | 2.8 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 33 | 3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | | 1 | | 1 | 1 |
| 34 | 3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 35 | 3.2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 36 | 3.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 37 | 3.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 0 | 1 | 0 | 1 |
| 23 | | | | | | | | | | | | | | | | |
| 224 | Totals | | 4 | 24 | 4 | 24 | 4 | 24 | 4 | 24 | 28 | 16 | 12 | 16 | 12 | 28 |
| 2 x BSTR | Link | Whole | 1.79% | 10.71% | 1.79% | 10.71% | 1.79% | 10.71% | 1.79% | 10.71% | 12.50% | 7.14% | 5.36% | 7.14% | 5.36% | 12.50% |
| 2 x BSTR | Link | Stage | 14.29% | 85.71% | 14.29% | 85.71% | 14.29% | 85.71% | 14.29% | 85.71% | 100.00% | 57.14% | 42.86% | 57.14% | 42.86% | 100.00% |
| 184 | Totals | | 2 | 21 | 2 | 21 | 2 | 21 | 2 | 21 | 23 | 14 | 9 | 14 | 9 | 23 |
| Max PTR | Link | Whole | 1.09% | 11.41% | 1.09% | 11.41% | 1.09% | 11.41% | 1.09% | 11.41% | 12.50% | 7.61% | 4.89% | 7.61% | 4.89% | 12.50% |
| Max PTR | Link | Stage | 8.70% | 91.30% | 8.70% | 91.30% | 8.70% | 91.30% | 8.70% | 91.30% | 100.00% | 60.87% | 39.13% | 60.87% | 39.13% | 100.00% |

## *Appendix C.2: Specific Link Analysis Technique Calculations for Network C2*

| Network C2 | Factors | | Links | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Trial | PTR | BSTR | 1_15 | 1_16 | 15_4 | 16_3 | 4_20 | 3_19 | 20_22 | 19_22 | 22_18 | 18_11 | 11_24 | 24_14 |
| 1 | 1.3 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 |
| 2 | 1.3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 3 | 1.3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 4 | 1.4 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 |
| 5 | 1.4 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 6 | 1.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 7 | 1.5 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 |
| 8 | 1.5 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 9 | 1.5 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 10 | 1.6 | 1.2 | 1 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 |
| 11 | 1.6 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 12 | 1.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 13 | 1.7 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 14 | 1.7 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 15 | 1.8 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 16 | 1.8 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 17 | 1.9 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 18 | 1.9 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 19 | 2 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 20 | 2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 21 | 2.1 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 22 | 2.1 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 23 | 2.2 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 24 | 2.2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 25 | 2.3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 26 | 2.3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 27 | 2.4 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 28 | 2.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 29 | 2.6 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 30 | 2.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 31 | 2.8 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 32 | 2.8 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 33 | 3 | 0.9 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 34 | 3 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 35 | 3.2 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 36 | 3.4 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 37 | 3.6 | 0.6 | | 1 | | 1 | | 1 | | 1 | 1 | 1 | 1 | 1 |
| 23 | | | | | | | | | | | | | | |
| 224 | Totals | | 4 | 24 | 4 | 24 | 4 | 24 | 4 | 24 | 28 | 28 | 28 | 28 |
| 2 x BSTR | Link | Whole | 1.79% | 10.71% | 1.79% | 10.71% | 1.79% | 10.71% | 1.79% | 10.71% | 12.50% | 12.50% | 12.50% | 12.50% |
| 2 x BSTR | Link | Stage | 14.29% | 85.71% | 14.29% | 85.71% | 14.29% | 85.71% | 14.29% | 85.71% | 100.00% | 100.00% | 100.00% | 100.00% |
| 184 | Totals | | 2 | 21 | 2 | 21 | 2 | 21 | 2 | 21 | 23 | 23 | 23 | 23 |
| Max PTR | Link | Whole | 1.09% | 11.41% | 1.09% | 11.41% | 1.09% | 11.41% | 1.09% | 11.41% | 12.50% | 12.50% | 12.50% | 12.50% |
| Max PTR | Link | Stage | 8.70% | 91.30% | 8.70% | 91.30% | 8.70% | 91.30% | 8.70% | 91.30% | 100.00% | 100.00% | 100.00% | 100.00% |

## *Appendix C.3: Additional Generated Routes for Network C2*

In Network C2, node 21's processing was only decreased by 12.5% to 1.75 from Network C1's original value of 2.00. Text output A shows the generated route without node 21. But, when node 21's processing was decreased by 25.0% to 1.50 from Network C1's original value of 2.00. Text output B shows the generated route using node 21 instead of using node 20.

TEXT OUTPUT A

```
PTR 1.60 PTRVar 1.00 BSTR 1.20 BSTRVar 1.00.
1 15 4 20 23 18 11 25 14
Expected sojourn time is 51.53200.
```

TEXT OUTPUT B

```
PTR 1.60 PTRVar 1.00 BSTR 1.20 BSTRVar 1.00.
1 15 4 21 23 18 11 25 14
Expected sojourn time is 51.03200.
```

## *Appendix D.1: Lower Bound Calculations for Network D (Varying Probability)*

| | | | | | |
|---|---|---|---|---|---|
| Network D-Lower Bound without delay | | | | | |
| # of Nodes | 4 | 4 | 4 | 4 | 4 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 5% | 10% | 20% | 30% | 40% |
| Probability (1-P) greater than Lower Bound | 95% | 90% | 80% | 70% | 60% |
| | | | | | |
| Standardized Variable Value | -1.64931 | -1.28128 | -0.83856 | -0.52154 | -0.25173 |
| | | | | | |
| Lower Bound Value | -5.19446 | -2.25023 | 1.291525 | 3.827664 | 5.986143 |
| | | | | | |
| Network D-Lower Bound with delay (Low) | | | | | |
| # of Nodes | 4 | 4 | 4 | 4 | 4 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 5% | 10% | 20% | 30% | 40% |
| Probability (1-P) greater than Lower Bound | 95% | 90% | 80% | 70% | 60% |
| | | | | | |
| Total Mean Sojourn Time | 8.6376 | 8.6376 | 8.6376 | 8.6376 | 8.6376 |
| Total Variance | 18.478 | 18.478 | 18.478 | 18.478 | 18.478 |
| Standardized Variable Value | -1.64931 | -1.28128 | -0.83856 | -0.52154 | -0.25173 |
| | | | | | |
| Lower Bound Value | 1.547882 | 3.129891 | 5.032965 | 6.395697 | 7.555503 |
| | | | | | |
| Network D-Lower Bound with delay (Medium) | | | | | |
| # of Nodes | 4 | 4 | 4 | 4 | 4 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 5% | 10% | 20% | 30% | 40% |
| Probability (1-P) greater than Lower Bound | 95% | 90% | 80% | 70% | 60% |
| | | | | | |
| Total Mean Sojourn Time | 15.9810 | 15.9810 | 15.9810 | 15.9810 | 15.9810 |
| Total Variance | 68.471 | 68.471 | 68.471 | 68.471 | 68.471 |
| Standardized Variable Value | -1.64931 | -1.28128 | -0.83856 | -0.52154 | -0.25173 |
| | | | | | |
| Lower Bound Value | 2.333446 | 5.37878 | 9.042155 | 11.66539 | 13.89799 |
| | | | | | |
| Network D-Lower Bound with delay (High) | | | | | |
| # of Nodes | 4 | 4 | 4 | 4 | 4 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 5% | 10% | 20% | 30% | 40% |
| Probability (1-P) greater than Lower Bound | 95% | 90% | 80% | 70% | 60% |
| | | | | | |
| Total Mean Sojourn Time | 77.0190 | 77.0190 | 77.0190 | 77.0190 | 77.0190 |
| Total Variance | 3654.8 | 3654.8 | 3654.8 | 3654.8 | 3654.8 |
| Standardized Variable Value | -1.64931 | -1.28128 | -0.83856 | -0.52154 | -0.25173 |
| | | | | | |
| Lower Bound Value | -22.6898 | -0.44061 | 26.32394 | 45.48921 | 61.80055 |

## *Appendix D.2: Lower Bound Calculations for Network D (Varying Nodes)*

| | | | | | |
|---|---|---|---|---|---|
| Network D-Lower Bound without delay | | | | | |
| # of Nodes | 4 | 8 | 10 | 20 | 40 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.83856 |
| | | | | | |
| Lower Bound Value | 1.291525 | 6.512783 | 9.392969 | 24.99939 | 58.78594 |
| | | | | | |
| Network D-Lower Bound with delay (Low) | | | | | |
| # of Nodes | 4 | 8 | 10 | 20 | 40 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 8.6376 | 17.2752 | 21.5940 | 43.1880 | 86.3760 |
| Total Variance | 18.478 | 18.478 | 18.478 | 18.478 | 18.478 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.83856 |
| | | | | | |
| Lower Bound Value | 5.032965 | 13.67056 | 17.98936 | 39.58336 | 82.77136 |
| | | | | | |
| Network D-Lower Bound with delay (Medium) | | | | | |
| # of Nodes | 4 | 8 | 10 | 20 | 40 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 15.9810 | 31.9620 | 39.9525 | 79.9050 | 159.8100 |
| Total Variance | 68.471 | 68.471 | 68.471 | 68.471 | 68.471 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.83856 |
| | | | | | |
| Lower Bound Value | 9.042155 | 25.02316 | 33.01366 | 72.96616 | 152.8712 |
| | | | | | |
| Network D-Lower Bound with delay (High) | | | | | |
| # of Nodes | 4 | 8 | 10 | 20 | 40 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 77.0190 | 154.0380 | 192.5475 | 385.0950 | 770.1900 |
| Total Variance | 3654.8 | 3654.8 | 3654.8 | 3654.8 | 3654.8 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.83856 |
| | | | | | |
| Lower Bound Value | 26.32394 | 103.3429 | 141.8524 | 334.3999 | 719.4949 |

## *Appendix D.3: Lower Bound Calculations for Network D (Doubling Nodes)*

| Network D-Lower Bound with delay (High) | | | | | |
|---|---|---|---|---|---|
| # of Nodes | 4 | 8 | 16 | 32 | 64 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 77.0190 | 154.0380 | 308.0760 | 616.1520 | 1232.3040 |
| Total Variance | 3654.8 | 3654.8 | 3654.8 | 3654.8 | 3654.8 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.838559 |
| | | | | | |
| High Traffic Level | 26.32394 | 103.3429 | 257.3809 | 565.4569 | 1181.6089 |
| | | | | | |
| Network D-Lower Bound with delay (Medium) | | | | | |
| # of Nodes | 4 | 8 | 16 | 32 | 64 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 15.9810 | 31.9620 | 63.9240 | 127.8480 | 255.6960 |
| Total Variance | 68.471 | 68.471 | 68.471 | 68.471 | 68.471 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.838559 |
| | | | | | |
| Medium Traffic Level | 9.042155 | 25.02316 | 56.98516 | 120.9092 | 248.75716 |
| | | | | | |
| # of Nodes | 4 | 8 | 16 | 32 | 64 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Total Mean Sojourn Time | 8.6376 | 17.2752 | 34.5504 | 69.1008 | 138.2016 |
| Total Variance | 18.478 | 18.478 | 18.478 | 18.478 | 18.478 |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.838559 |
| | | | | | |
| Low Traffic Level | 5.032965 | 13.67056 | 30.94576 | 65.49616 | 134.59696 |
| | | | | | |
| | | | | | |
| Network D-Lower Bound without delay | | | | | |
| # of Nodes | 4 | 8 | 16 | 32 | 64 |
| Mean Service Time | 2 | 2 | 2 | 2 | 2 |
| Variance | 4 | 4 | 4 | 4 | 4 |
| Probability P less than Lower Bound | 20% | 20% | 20% | 20% | 20% |
| Probability (1-P) greater than Lower Bound | 80% | 80% | 80% | 80% | 80% |
| | | | | | |
| Standardized Variable Value | -0.83856 | -0.83856 | -0.83856 | -0.83856 | -0.838559 |
| | | | | | |
| No Queueing | 1.291525 | 6.512783 | 18.58305 | 45.02557 | 101.1661 |

## *Appendix D.4: Constant Growth Rate (Low Traffic)*

| Lower Bound | # of Nodes | Mean Service Time | Std Dev. | Total Delay Time | Probability | [Klo96] |
|---|---|---|---|---|---|---|
| 1.2582 | 1 | 2.1594 | 4.298604425 | 2.1594 | 0.5830 | 1.2641 |
| 2.5165 | 2 | 2.1594 | 4.298604425 | 4.3188 | 0.6625 | 2.5271 |
| 3.7747 | 3 | 2.1594 | 4.298604425 | 6.4782 | 0.7353 | 3.7880 |
| 5.0329 | 4 | 2.1594 | 4.298604425 | 8.6376 | 0.7991 | 5.0461 |
| 6.2911 | 5 | 2.1594 | 4.298604425 | 10.7970 | 0.8527 | 6.3007 |
| 7.5494 | 6 | 2.1594 | 4.298604425 | 12.9564 | 0.8958 | 7.5516 |
| 8.8076 | 7 | 2.1594 | 4.298604425 | 15.1158 | 0.9289 | 8.7992 |
| 10.0658 | 8 | 2.1594 | 4.298604425 | 17.2752 | 0.9532 | 10.0446 |
| 11.3240 | 9 | 2.1594 | 4.298604425 | 19.4346 | 0.9704 | 11.2899 |
| 12.5823 | 10 | 2.1594 | 4.298604425 | 21.5940 | 0.9820 | 12.5382 |
| 13.8405 | 11 | 2.1594 | 4.298604425 | 23.7534 | 0.9894 | 13.7933 |
| 15.0987 | 12 | 2.1594 | 4.298604425 | 25.9128 | 0.9941 | 15.0599 |
| 16.3569 | 13 | 2.1594 | 4.298604425 | 28.0722 | 0.9968 | 16.3435 |
| 17.6152 | 14 | 2.1594 | 4.298604425 | 30.2316 | 0.9983 | 17.6494 |
| 18.8734 | 15 | 2.1594 | 4.298604425 | 32.3910 | 0.9992 | 18.9832 |
| 20.1316 | 16 | 2.1594 | 4.298604425 | 34.5504 | 0.9996 | 20.3501 |
| 21.3898 | 17 | 2.1594 | 4.298604425 | 36.7098 | 0.9998 | 21.7545 |
| 22.6481 | 18 | 2.1594 | 4.298604425 | 38.8692 | 0.9999 | 23.2002 |
| 23.9063 | 19 | 2.1594 | 4.298604425 | 41.0286 | 1.0000 | 24.6902 |
| 25.1645 | 20 | 2.1594 | 4.298604425 | 43.1880 | 1.0000 | 26.2265 |
| 26.4227 | 21 | 2.1594 | 4.298604425 | 45.3474 | 1.0000 | 27.8104 |
| 27.6810 | 22 | 2.1594 | 4.298604425 | 47.5068 | 1.0000 | 29.4421 |
| 28.9392 | 23 | 2.1594 | 4.298604425 | 49.6662 | 1.0000 | 31.1212 |
| 30.1974 | 24 | 2.1594 | 4.298604425 | 51.8256 | 1.0000 | 32.8467 |
| 31.4556 | 25 | 2.1594 | 4.298604425 | 53.9850 | 1.0000 | 34.6169 |
| 32.7139 | 26 | 2.1594 | 4.298604425 | 56.1444 | 1.0000 | 36.4296 |
| 33.9721 | 27 | 2.1594 | 4.298604425 | 58.3038 | 1.0000 | 38.2824 |
| 35.2303 | 28 | 2.1594 | 4.298604425 | 60.4632 | 1.0000 | 40.1725 |
| 36.4885 | 29 | 2.1594 | 4.298604425 | 62.6226 | 1.0000 | 42.0970 |
| 37.7468 | 30 | 2.1594 | 4.298604425 | 64.7820 | 1.0000 | 44.0529 |
| 39.0050 | 31 | 2.1594 | 4.298604425 | 66.9414 | 1.0000 | 46.0372 |
| 40.2632 | 32 | 2.1594 | 4.298604425 | 69.1008 | 1.0000 | 48.0470 |
| 41.5214 | 33 | 2.1594 | 4.298604425 | 71.2602 | 1.0000 | 50.0794 |
| 42.7797 | 34 | 2.1594 | 4.298604425 | 73.4196 | 1.0000 | 52.1317 |
| 44.0379 | 35 | 2.1594 | 4.298604425 | 75.5790 | 1.0000 | 54.2013 |
| 45.2961 | 36 | 2.1594 | 4.298604425 | 77.7384 | 1.0000 | 56.2860 |
| 46.5543 | 37 | 2.1594 | 4.298604425 | 79.8978 | 1.0000 | 58.3840 |
| 47.8126 | 38 | 2.1594 | 4.298604425 | 82.0572 | 1.0000 | 60.4907 |
| 49.0708 | 39 | 2.1594 | 4.298604425 | 84.2166 | 1.0000 | 62.6042 |
| 50.3290 | 40 | 2.1594 | 4.298604425 | 86.3760 | 1.0000 | 64.6109 |

## *Appendix D.5: Constant Growth Rate (Medium Traffic)*

| Lower Bound | # of Nodes | Mean Service Time | Std Dev. | Total Delay Time | Probability | [Klo96] |
|---|---|---|---|---|---|---|
| 2.2605 | 1 | 3.99525 | 8.274720539 | 3.9953 | 0.5830 | 2.2718 |
| 4.5211 | 2 | 3.99525 | 8.274720539 | 7.9905 | 0.6625 | 4.5415 |
| 6.7816 | 3 | 3.99525 | 8.274720539 | 11.9858 | 0.7353 | 6.8073 |
| 9.0422 | 4 | 3.99525 | 8.274720539 | 15.9810 | 0.7991 | 9.0675 |
| 11.3027 | 5 | 3.99525 | 8.274720539 | 19.9763 | 0.8527 | 11.3211 |
| 13.5632 | 6 | 3.99525 | 8.274720539 | 23.9715 | 0.8958 | 13.5675 |
| 15.8238 | 7 | 3.99525 | 8.274720539 | 27.9668 | 0.9289 | 15.8076 |
| 18.0843 | 8 | 3.99525 | 8.274720539 | 31.9620 | 0.9532 | 18.0435 |
| 20.3448 | 9 | 3.99525 | 8.274720539 | 35.9573 | 0.9704 | 20.2793 |
| 22.6054 | 10 | 3.99525 | 8.274720539 | 39.9525 | 0.9820 | 22.5206 |
| 24.8659 | 11 | 3.99525 | 8.274720539 | 43.9478 | 0.9894 | 24.7750 |
| 27.1265 | 12 | 3.99525 | 8.274720539 | 47.9430 | 0.9941 | 27.0518 |
| 29.3870 | 13 | 3.99525 | 8.274720539 | 51.9383 | 0.9968 | 29.3611 |
| 31.6475 | 14 | 3.99525 | 8.274720539 | 55.9335 | 0.9983 | 31.7135 |
| 33.9081 | 15 | 3.99525 | 8.274720539 | 59.9288 | 0.9992 | 34.1195 |
| 36.1686 | 16 | 3.99525 | 8.274720539 | 63.9240 | 0.9996 | 36.5891 |
| 38.4292 | 17 | 3.99525 | 8.274720539 | 67.9193 | 0.9998 | 39.1310 |
| 40.6897 | 18 | 3.99525 | 8.274720539 | 71.9145 | 0.9999 | 41.7524 |
| 42.9502 | 19 | 3.99525 | 8.274720539 | 75.9098 | 1.0000 | 44.4590 |
| 45.2108 | 20 | 3.99525 | 8.274720539 | 79.9050 | 1.0000 | 47.2549 |
| 47.4713 | 21 | 3.99525 | 8.274720539 | 83.9003 | 1.0000 | 50.1422 |
| 49.7319 | 22 | 3.99525 | 8.274720539 | 87.8955 | 1.0000 | 53.1217 |
| 51.9924 | 23 | 3.99525 | 8.274720539 | 91.8908 | 1.0000 | 56.1924 |
| 54.2529 | 24 | 3.99525 | 8.274720539 | 95.8860 | 1.0000 | 59.3523 |
| 56.5135 | 25 | 3.99525 | 8.274720539 | 99.8813 | 1.0000 | 62.5983 |
| 58.7740 | 26 | 3.99525 | 8.274720539 | 103.8765 | 1.0000 | 65.9262 |
| 61.0345 | 27 | 3.99525 | 8.274720539 | 107.8718 | 1.0000 | 69.3312 |
| 63.2951 | 28 | 3.99525 | 8.274720539 | 111.8670 | 1.0000 | 72.8081 |
| 65.5556 | 29 | 3.99525 | 8.274720539 | 115.8623 | 1.0000 | 76.3511 |
| 67.8162 | 30 | 3.99525 | 8.274720539 | 119.8575 | 1.0000 | 79.9547 |
| 70.0767 | 31 | 3.99525 | 8.274720539 | 123.8528 | 1.0000 | 83.6129 |
| 72.3372 | 32 | 3.99525 | 8.274720539 | 127.8480 | 1.0000 | 87.3201 |
| 74.5978 | 33 | 3.99525 | 8.274720539 | 131.8433 | 1.0000 | 91.0708 |
| 76.8583 | 34 | 3.99525 | 8.274720539 | 135.8385 | 1.0000 | 94.8598 |
| 79.1189 | 35 | 3.99525 | 8.274720539 | 139.8338 | 1.0000 | 98.6822 |
| 81.3794 | 36 | 3.99525 | 8.274720539 | 143.8290 | 1.0000 | 102.5336 |
| 83.6399 | 37 | 3.99525 | 8.274720539 | 147.8243 | 1.0000 | 106.4107 |
| 85.9005 | 38 | 3.99525 | 8.274720539 | 151.8195 | 1.0000 | 110.3045 |
| 88.1610 | 39 | 3.99525 | 8.274720539 | 155.8148 | 1.0000 | 114.2114 |
| 90.4216 | 40 | 3.99525 | 8.274720539 | 159.8100 | 1.0000 | 117.9127 |

## *Appendix D.6: Constant Growth Rate (High Traffic)*

| Lower Bound | # of Nodes | Mean Service Time | Std Dev. | Total Delay Time | Probability | [Klo96] |
|---|---|---|---|---|---|---|
| 6.5809 | 1 | 19.25475 | 60.4549419 | 19.2548 | 0.5830 | 6.6628 |
| 13.1617 | 2 | 19.25475 | 60.4549419 | 38.5095 | 0.6625 | 13.3108 |
| 19.7426 | 3 | 19.25475 | 60.4549419 | 57.7643 | 0.7353 | 19.9302 |
| 26.3239 | 4 | 19.25475 | 60.4549419 | 77.0190 | 0.7991 | 26.5094 |
| 32.9043 | 5 | 19.25475 | 60.4549419 | 96.2738 | 0.8527 | 33.0385 |
| 39.4851 | 6 | 19.25475 | 60.4549419 | 115.5285 | 0.8958 | 39.5164 |
| 46.0660 | 7 | 19.25475 | 60.4549419 | 134.7833 | 0.9289 | 45.9478 |
| 52.6468 | 8 | 19.25475 | 60.4549419 | 154.0380 | 0.9532 | 52.3489 |
| 59.2277 | 9 | 19.25475 | 60.4549419 | 173.2928 | 0.9704 | 58.7484 |
| 65.8085 | 10 | 19.25475 | 60.4549419 | 192.5475 | 0.9820 | 65.1887 |
| 72.3894 | 11 | 19.25475 | 60.4549419 | 211.8023 | 0.9894 | 71.7253 |
| 78.9702 | 12 | 19.25475 | 60.4549419 | 231.0570 | 0.9941 | 78.4249 |
| 85.5511 | 13 | 19.25475 | 60.4549419 | 250.3118 | 0.9968 | 85.3618 |
| 92.1319 | 14 | 19.25475 | 60.4549419 | 269.5665 | 0.9983 | 92.6137 |
| 98.7128 | 15 | 19.25475 | 60.4549419 | 288.8213 | 0.9992 | 100.2578 |
| 105.2936 | 16 | 19.25475 | 60.4549419 | 308.0760 | 0.9996 | 108.3660 |
| 111.8745 | 17 | 19.25475 | 60.4549419 | 327.3308 | 0.9998 | 117.0022 |
| 118.4553 | 18 | 19.25475 | 60.4549419 | 346.5855 | 0.9999 | 126.2198 |
| 125.0362 | 19 | 19.25475 | 60.4549419 | 365.8403 | 1.0000 | 136.0602 |
| 131.6170 | 20 | 19.25475 | 60.4549419 | 385.0950 | 1.0000 | 146.5522 |
| 138.1979 | 21 | 19.25475 | 60.4549419 | 404.3498 | 1.0000 | 157.7125 |
| 144.7787 | 22 | 19.25475 | 60.4549419 | 423.6045 | 1.0000 | 169.5458 |
| 151.3596 | 23 | 19.25475 | 60.4549419 | 442.8593 | 1.0000 | 182.0462 |
| 157.9404 | 24 | 19.25475 | 60.4549419 | 462.1140 | 1.0000 | 195.1984 |
| 164.5213 | 25 | 19.25475 | 60.4549419 | 481.3688 | 1.0000 | 208.9790 |
| 171.1021 | 26 | 19.25475 | 60.4549419 | 500.6235 | 1.0000 | 223.3581 |
| 177.6830 | 27 | 19.25475 | 60.4549419 | 519.8783 | 1.0000 | 238.3007 |
| 184.2638 | 28 | 19.25475 | 60.4549419 | 539.1330 | 1.0000 | 253.7682 |
| 190.8447 | 29 | 19.25475 | 60.4549419 | 558.3878 | 1.0000 | 269.7195 |
| 197.4255 | 30 | 19.25475 | 60.4549419 | 577.6425 | 1.0000 | 286.1124 |
| 204.0064 | 31 | 19.25475 | 60.4549419 | 596.8973 | 1.0000 | 302.9048 |
| 210.5872 | 32 | 19.25475 | 60.4549419 | 616.1520 | 1.0000 | 320.0551 |
| 217.1681 | 33 | 19.25475 | 60.4549419 | 635.4068 | 1.0000 | 337.5232 |
| 223.7489 | 34 | 19.25475 | 60.4549419 | 654.6615 | 1.0000 | 355.2713 |
| 230.3298 | 35 | 19.25475 | 60.4549419 | 673.9163 | 1.0000 | 373.2639 |
| 236.9106 | 36 | 19.25475 | 60.4549419 | 693.1710 | 1.0000 | 391.4676 |
| 243.4915 | 37 | 19.25475 | 60.4549419 | 712.4258 | 1.0000 | 409.8588 |
| 250.0723 | 38 | 19.25475 | 60.4549419 | 731.6805 | 1.0000 | 428.3726 |
| 256.6532 | 39 | 19.25475 | 60.4549419 | 750.9353 | 1.0000 | 446.9821 |
| 263.2340 | 40 | 19.25475 | 60.4549419 | 770.1900 | 1.0000 | 464.0890 |

## *Appendix D.6: Upper Bound Calculations for Network D (All Trials)*

| | | | | | |
|---|---|---|---|---|---|
| Network D-Upper Bound of High Intensity Trial | | | | | |
| Quantile q less than Upper Bound | 90 | 90 | 90 | 90 | |
| Quantile (1-q) greater than Upper Bound | 10 | 10 | 10 | 10 | |
| Node Sequence number | 1 | 2 | 3 | 4 | |
| Mean Servive Rate | 0.5 | 0.5 | 0.5 | 0.5 | |
| Node traffic intensity | 0.72 | 0.72 | 0.96 | 0.36 | |
| | | | | | |
| Upper bound per node | 30.54762 | 30.54762 | 228.2174 | 11.1985 | |
| Upper Bound Value for primary customer's path. | | | | | 300.5111 |
| | | | | | |
| Network D-Upper Bound of Medium Intensity Trial | | | | | |
| Quantile q less than Upper Bound | 90 | 90 | 90 | 90 | |
| Quantile (1-q) greater than Upper Bound | 10 | 10 | 10 | 10 | |
| Node Sequence number | 1 | 2 | 3 | 4 | |
| Mean Servive Rate | 0.5 | 0.5 | 0.5 | 0.5 | |
| Node traffic intensity | 0.44 | 0.48 | 0.621 | 0.24 | |
| | | | | | |
| Upper bound per node | 13.51496 | 14.88923 | 21.78758 | 8.3633 | |
| Upper Bound Value for primary customer's path. | | | | | 58.55507 |
| | | | | | |
| Network D-Upper Bound of Low Intensity Trial | | | | | |
| Quantile q less than Upper Bound | 90 | 90 | 90 | 90 | |
| Quantile (1-q) greater than Upper Bound | 10 | 10 | 10 | 10 | |
| Node Sequence number | 1 | 2 | 3 | 4 | |
| Mean Servive Rate | 0.5 | 0.5 | 0.5 | 0.5 | |
| Node traffic intensity | 0.06 | 0.08 | 0.096 | 0.04 | |
| | | | | | |
| Upper bound per node | 3.812254 | 4.520525 | 5.003901 | 2.888113 | |
| Upper Bound Value for primary customer's path. | | | | | 16.22479 |

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

| | | |
|---|---|---|
| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE**<br>December 1996 | **3. REPORT TYPE AND DATES COVERED**<br>Master's Thesis |

**4. TITLE AND SUBTITLE**
A SPECIFIC NETWORK LINK AND PATH LIKELIHOOD PREDICTION TOOL

**5. FUNDING NUMBERS**

**6. AUTHOR(S)**
Gary K. Moy, Captain, USAF

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Air Force Institute of Technology,
2950 P Street
WPAFB OH 45433-7765

**8. PERFORMING ORGANIZATION REPORT NUMBER**
AFIT/GCS/ENG/96D-21

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Dept of Defense (R55)
9800 Savage Road, Ste 6550
Fort Meade, MD 20755-6000

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12a. DISTRIBUTION/AVAILABILITY STATEMENT**
Approved for public release; distribution unlimited

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

In this study, Dijkstra's algorithm has been modified to allow the Queueing Network Analyzer's (QNA) analysis output to act as a node's goodness metric. QNA's calculation of the expected Sojourn Time in a node provides accurate measurement of expected congestion. The modified Dijkstra's algorithm in the Generalized Network Analyzer (GNA) is verified and empirically validated to properly deliver traffic from start to destination. GNA's Congestion Control displays notification and informs the user certain network input parameters must be lowered or where certain nodes must be improved to maintain node stability. Upon successful completion of the analysis, GNA outputs the generated route and expected sojourn time for later analysis. Use of two analysis techniques show the percentage of link usage within a 25 node test network. Three analytical techniques are provided to estimate the probable bounds of the input parameters and sojourn times. Using these techniques, a bound of the Total Sojourn Times is provided for a 16 node test network. Given few input parameters, networks analyzed can provide a specific link usage probability and path likelihood. Since QNA requires few calculations and GNA's Congestion Control provides unstable node identification, designers and engineers can evaluate network topologies much more easily.

**14. SUBJECT TERMS**
Generalized Network Analyzer, GNA, Queueing Network Analyzer, QNA, Whitt
Communication Networks, Computer Networks, Topology, Queueing Theory

**15. NUMBER OF PAGES**
175

**16. PRICE CODE**

| | | | |
|---|---|---|---|
| **17. SECURITY CLASSIFICATION OF REPORT** | **18. SECURITY CLASSIFICATION OF THIS PAGE** | **19. SECURITY CLASSIFICATION OF ABSTRACT** | **20. LIMITATION OF ABSTRACT** |
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |

# GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements***.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

| | | | | | |
|---|---|---|---|---|---|
| C | - | Contract | PR | - | Project |
| G | - | Grant | TA | - | Task |
| PE | - | Program Element | WU | - | Work Unit Accession No. |

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE - See authorities.
NASA - See Handbook NHB 2200.2.
NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.
DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
NASA - Leave blank.
NTIS - Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.